NeuroScholar Core 1.4

User Manual

(Build 1.4.001)

Knowledge Mechanics Research Group
University of Southern California
(2005)

Contents

NEUROSCHOLAR CORE 1.4 USER MANUAL 1

1. INTRODUCTION 3
2. WHAT YOU CAN DO WITH NEUROSCHOLAR CORE 1.4 4
3. THE NEUROSCHOLAR LICENSE 5
4. SYSTEM SET-UP 5
   4.1. Release 0.001, some additional installation and setup required. *5*
5. A QUICK TOUR THROUGH THE SYSTEM 6
   5.1. The System States *6*
      5.1.1. 'Home' 7
      5.1.2. 'Documentation' 9
      5.1.3. 'Query' 10
      5.1.4. 'List' 11
      5.1.5. 'Display' 13
      5.1.6. 'Insert' 14
      5.1.7. 'Edit' 14
   5.2. Plugins *14*
   5.3. 'Knowledge Browser' Toolbar commands *15*
   5.4. 'History' commands *16*
   5.5. 'Graph' commands *17*
      5.5.1. The Graph Panel – Definition Pane 18
      5.5.2. The Graph Panel – Instance Pane 21
      5.5.3. Visibility Controls 24
         5.5.3.1. View Definition Visibility 24
         5.5.3.2. View Instance Visibility 25
         5.5.3.3. Profiles 26
      5.5.4. An aside: the substructure of a view 27
   5.6. 'Tree' commands *28*
6. SYSTEM ADMINISTRATION HOW-TOS 30
   6.1. Creating a new knowledge base *30*
   6.2. Destroying a knowledge base *31*
   6.3. Saving a knowledge base *31*
   6.4. Restoring a saved knowledge base *32*
   6.5. Managing Users *34*
7. THE 'CORE' SYSTEM 35
   7.1. System packages *35*
8. HOW-TOS 35
   8.1. Enter a bibliographic reference into the system using PubMed *36*
   8.2. Organize papers using a ResourceCollection *37*
   8.3. Upload a library of citations from an EndNote, Reference Manager or Procite library *40*
   8.4. Add a PDF file to the reference *42*
   8.5. Locate an Article in the system *42*
   8.6. Fragment a PDF file *43*
   8.7. Insert a fragment with annotations (either text, voice) *45*
   8.8. Insert a fragment with data annotations *46*
   8.9. Display a fragment *47*
   8.10. Delete a fragment *47*
   8.11. Delete an article with all contained fragments *48*
   8.12. Linking up fragments with 'supports', or 'contradicts' relations *48*
   8.13. Inserting unattached Knowledge Statements *49*
9. ADDITIONAL SUPPORT 49

1. Introduction

The discipline of neuroinformatics can be thought of as 'the information science infrastructure of neuroscience' (Ascoli et al 2003). This usually involves the development of new software and databases to manage and publish neuroscientific data. Neuroscience already has a robust existing information science infrastructure: the neuroscientific literature.

It has been argued that science may be thought of as a *process of inscription*, meaning that the fundamental process of performing scientific work is simply that of generating inscriptions and then transforming those inscriptions from one form to another according to carefully-defined rules (Latour and Woolgar 1978). If this is the case, then the literature is the ultimate 'end-product' of most scientific work; it is the repository into which the majority of scientific knowledge is deposited; it is the currency through which scientific value is judged and it is also the crucible in which discoveries and contradictions are formally aired and elucidated. For this reason, the neuroscientific literature is the primary target for the action of the NeuroScholar system.

The basic schematic organization of the NeuroScholar system is shown in Figure 1. Quite simply, the literature exists as a large number of disparate, scattered 'portable document format' ('PDF') files that may be referenced via a set of uniquely definable keys (from online sources such as PubMed or each document's DOI address). NeuroScholar users may delineate fragments form within these publications to form individual facts. Once this step has been completed, users may then compile fragments together to support interpretations and then may build relations between those interpretations. When applied to specific, typed data in a well-defined ontology, this schema defines the basic design of the NeuroScholar system.

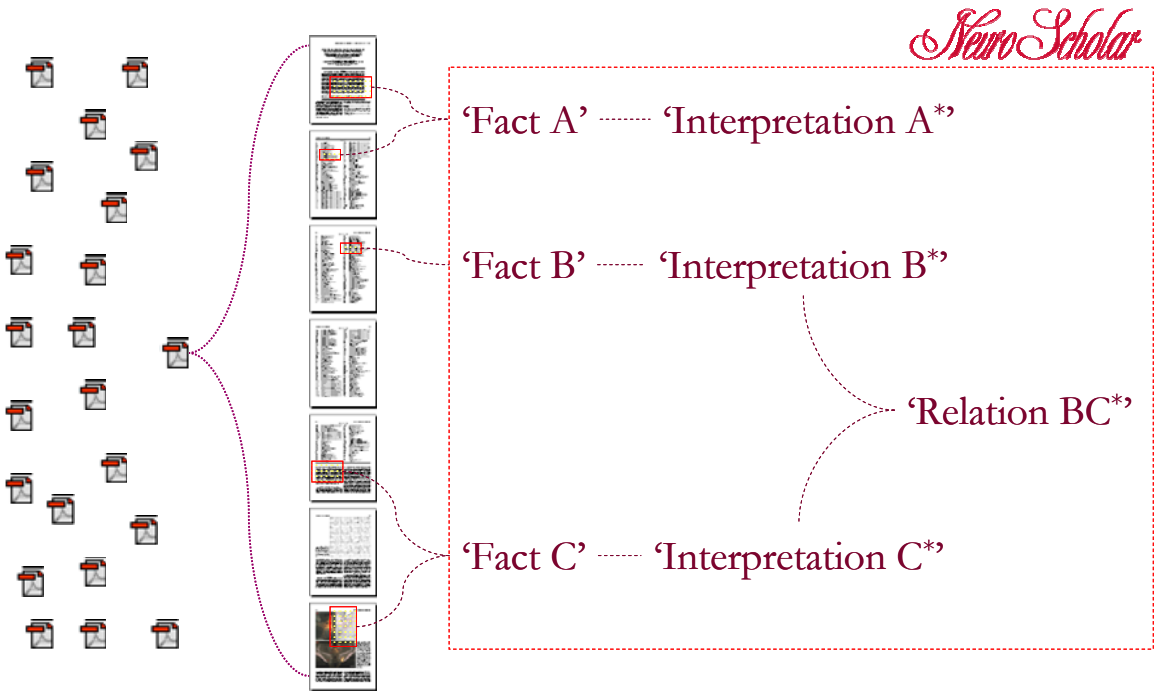


Figure 1: the design of the core model in the NeuroScholar system.

This approach may be generalized to any disparate source of information that could be of interest to the construction of a knowledge representation within neuroscience. A very wide range of data is available to neuroscientists from which they construct their understanding of the subject. This includes their notebook pages, posters at conferences, powerpoint presentations, images and data files downloaded from the web. All of these resources, in theory, can contribute to their knowledge and all of them could theoretically contribute to the construction of knowledge models. This, and other functionality, has been built into more the more advanced versions of NeuroScholar that have programming structure built on the foundations of the core system.

2. What you can do with NeuroScholar Core 1.4

NeuroScholar is primarily a system for managing your own knowledge. Most neuroscientists read a very large number of papers on a daily basis. NeuroScholar permits users to create a Knowledge Base (KB) of facts, based on the published literature that directly links to the primary literature itself. These facts may then be linked together to construct an argument. These facts take the

form of annotations over the literature using freeform text, voice recordings, or structured attribute-value pairs. Users may also use download references and PDF files rapidly from within the application to make interacting with the literature as straightforward as possible.

3. The NeuroScholar License

NeuroScholar is open-source software, meaning that we make the source code available for perusal, modification and redistribution where required. We have adapted the GNU General Public License stating that any work derived from NeuroScholar must also be distributed as an open source project. This is quite restrictive, but is suitable for academic work, ensuring that our software solution supports open sharing and distribution of source code.

4. System set-up

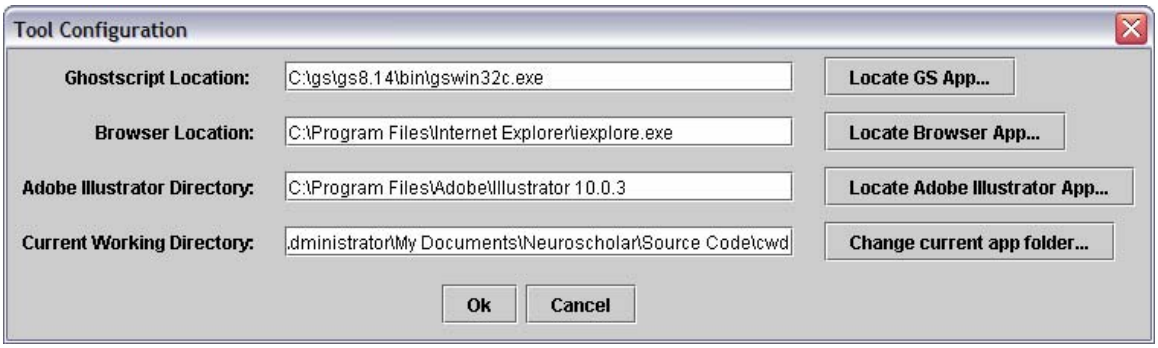
For the first time, the system is now available as an installation package. Users need only download the package, double-click it and then follow instructions to have NeuroScholar operational on their system.

Note that you must also install MySQL and you must know the root login and password.

4.1. Release 0.001, some additional installation and setup required.

Due to a minor bug, it is currently necessary to download and install GhostScript to ensure full system functionality. Ghostscript may be downloaded from http://www.cs.wisc.edu/~ghost/doc/AFPL. On running the system for the first time, users should configure the system selecting `File > Configure…`

A dialog box will pop up with the following structure.



To ensure proper system functionality, you need to tell NeuroScholar where it can find the executable files for Ghostscript (on Windows, this is a file called gswin32c.exe, on Mac OS X it is usually '/usr/local/bin/gs'), for a suitable Internet Browser (such as Internet Explorer on a PC or Safari on a Mac). This also permits you to define the current working directory's location which is where NeuroScholar will make local copies of PDF files and other required resources (such as temporary files and log files).

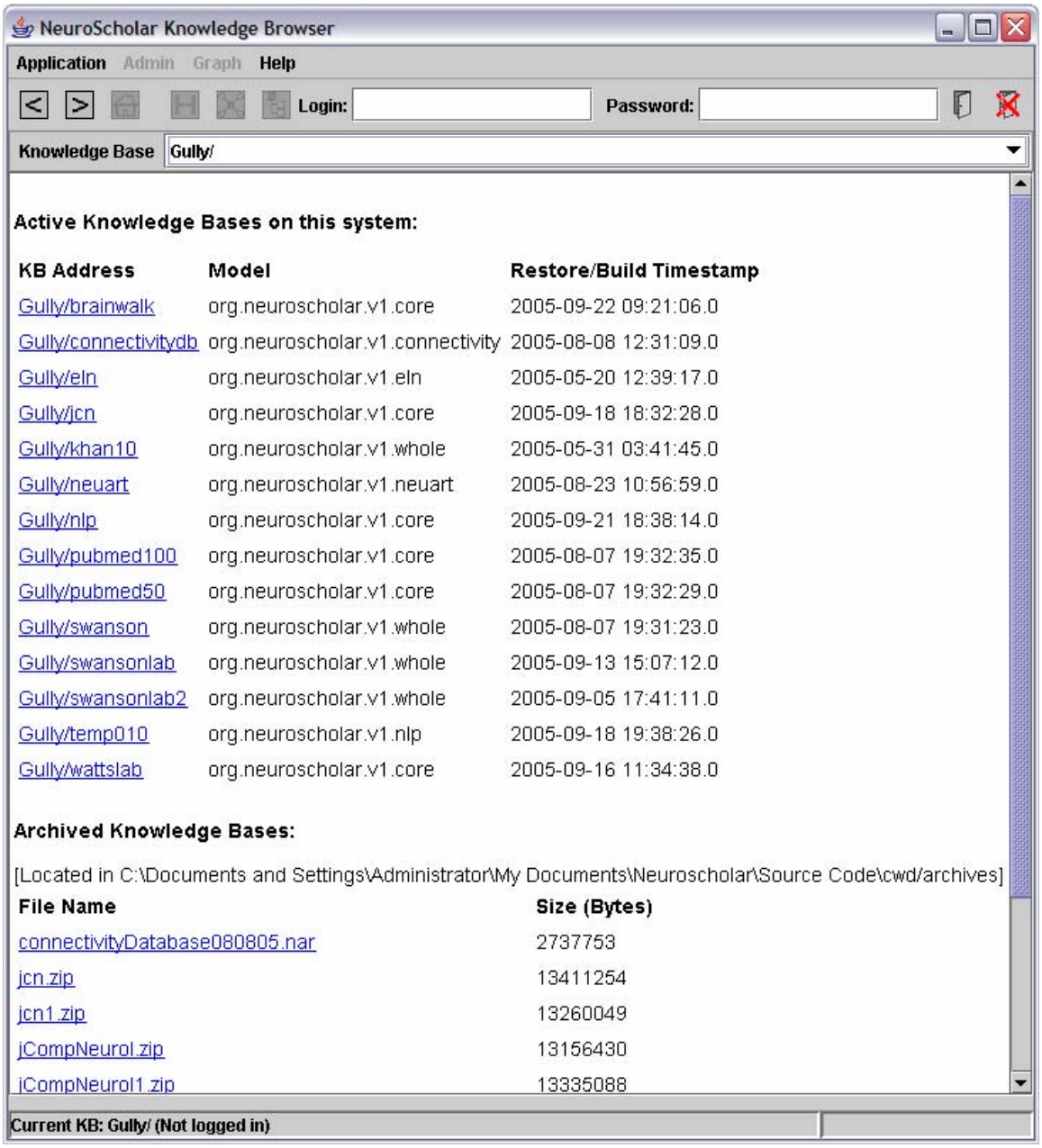
5. A quick tour through the system

5.1. The System States

NeuroScholar is essentially a database application that operates on specific views by being in any one of seven states at any time. Each view represents a specific class of data object in the system, such as a journal article, a fragment contained in an article or a statement describing some fact in the system. We will walkthrough the simple functionality of the system to illustrate how it is used and to display screenshots illustrating each state of the system.

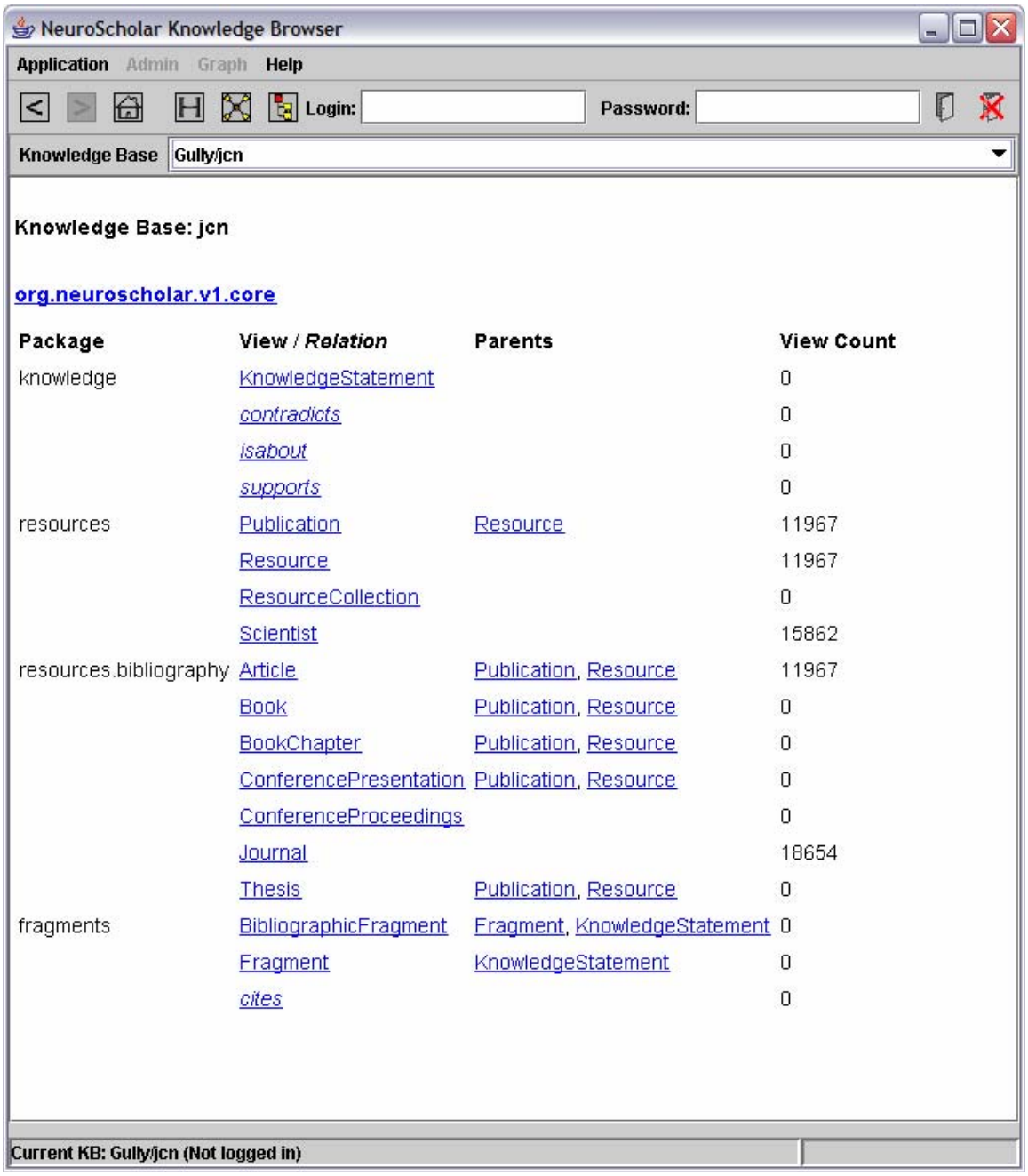
5.1.1. 'Home'

In this state, the system displays summary information for either the whole system or for a specific KB. When you first run the system, the first thing you see is shown below:



Each of the links in the top half of the screen represents a different KB (with the type of model supporting its construction and the date of its creation

listed). Note that there are many types of model; this manual is only concerned with those of type 'org.neuroscholar.v1.core' representing the core functionality of the system. The bottom half of the screen contains NeuroScholar archive files that contain all the data from a specific knowledge model. A user may restore the data from one of these archives by clicking on the link to the file. If a user selects one of the KBs on the system, he/she is taken to the following screen:



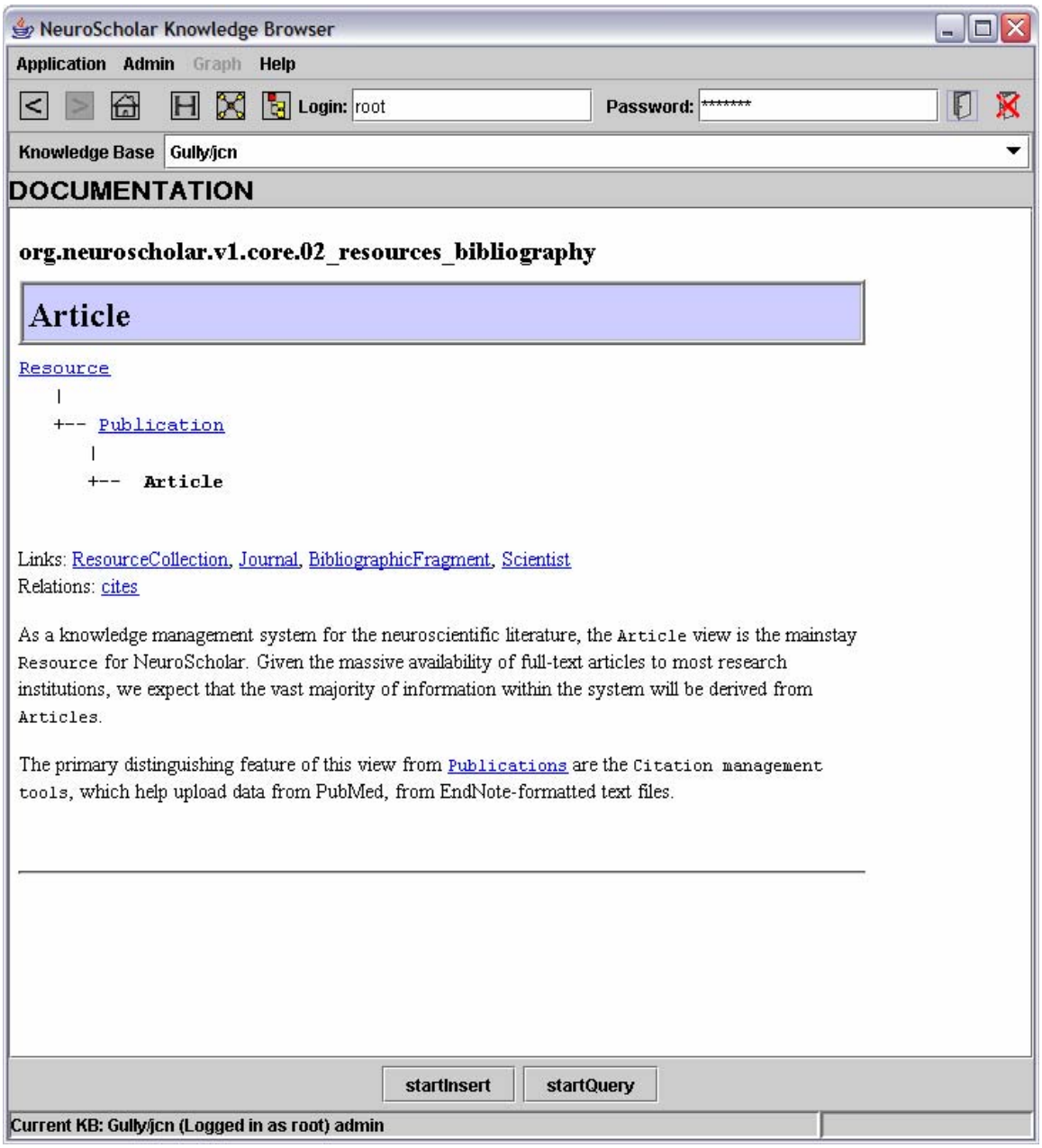
This screen lists all the different views and available relations in the selected knowledge base with counts of the number of instances of these views. Notice that the system comes preloaded with 18654 instances of the 'Journal' view, taken from PubMed's online database of journals.

The KB displayed here is a restored version of the '**jCompNeurol-1970-2005.nar**' NeuroScholar archive file that is distributed with the NeuroScholar Core 1.4 system. This knowledge base contains citation information from every issue of the Journal of Comparative Neurology from 1970 to 2005 in our attempt to make a definitive knowledge model of the literature underlying tract-tracing experiments. For this purposes of this distribution, it serves as an excellent example for new users of the system.

If the user clicks on any of the names of a View such as 'Article' (for example) they will be taken to the 'documentation' state.

5.1.2. 'Documentation'

In this state, the system displays some documentation describing the view. This documentation describes any Parent views of the view in question as well as links, dependencies and children where necessary. Clicking on links permits the user to easily navigate to those views' documentation as well.

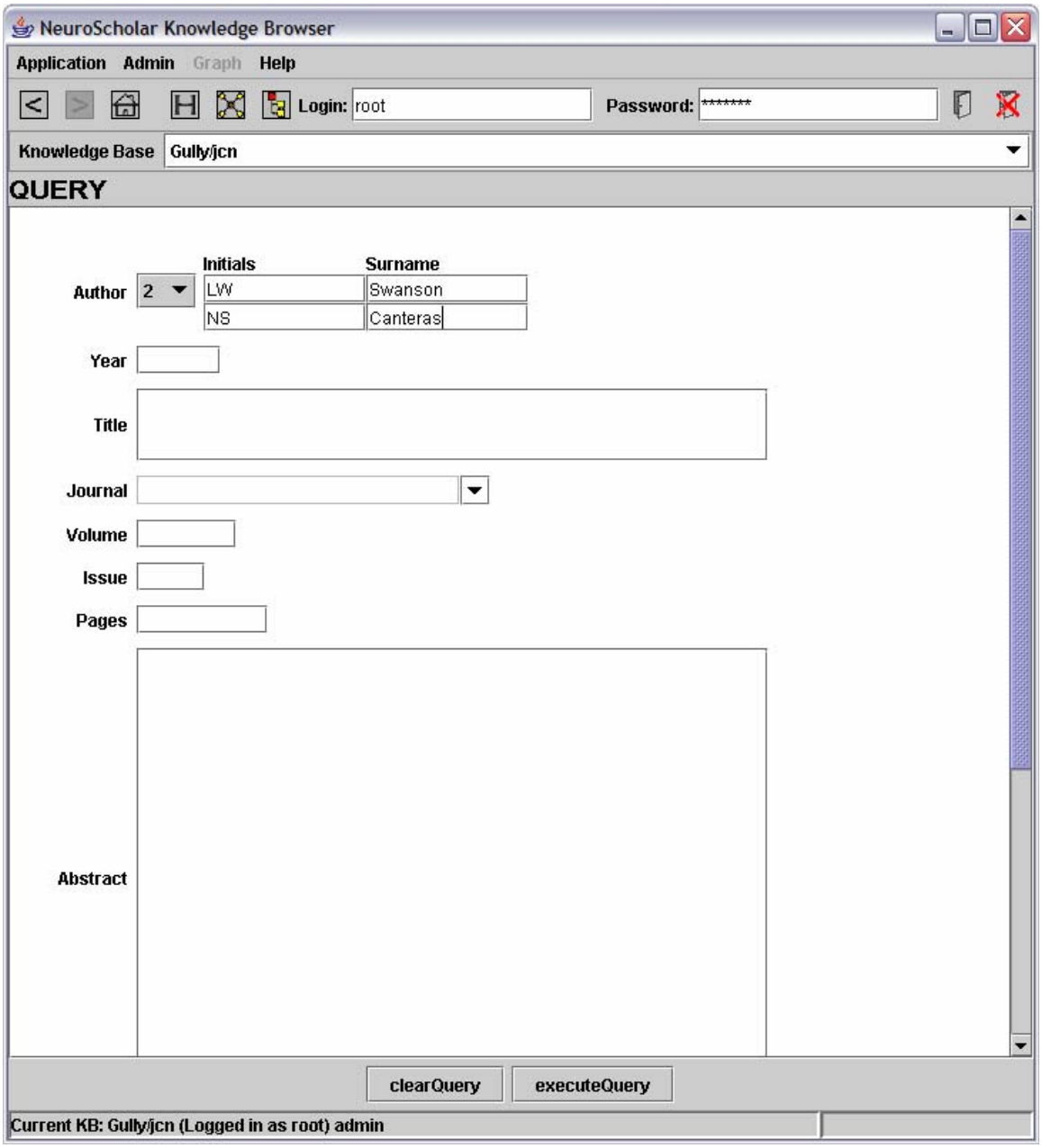

NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/jcn
DOCUMENTATION
org.neuroscholar.v1.core.02_resources_bibliography
Article
Resource
Publication
Article
Links: ResourceCollection, Journal, BibliographicFragment, Scientist
Relations: cites
As a knowledge management system for the neuroscientific literature, the Article view is the mainstay Resource for NeuroScholar. Given the massive availability of full-text articles to most research institutions, we expect that the vast majority of information within the system will be derived from Articles.
The primary distinguishing feature of this view from Publications are the Citation management tools, which help upload data from PubMed, from EndNote-formatted text files.
startInsert
startQuery
Current KB: Gully/jcn (Logged in as root) admin

Two buttons appear, providing additional functionality. We will consider the 'startInsert' button later. If we click 'startQuery', we enter the Query state.

5.1.3. 'Query'

When entering the query state, you will see a blank form where you may enter criteria for the attributes you wish to search for in the KB. You may use the "%" character as a wildcard. Some controls (such as for Author, shown below) have pull-down boxes that permit multiple criteria to be specified so

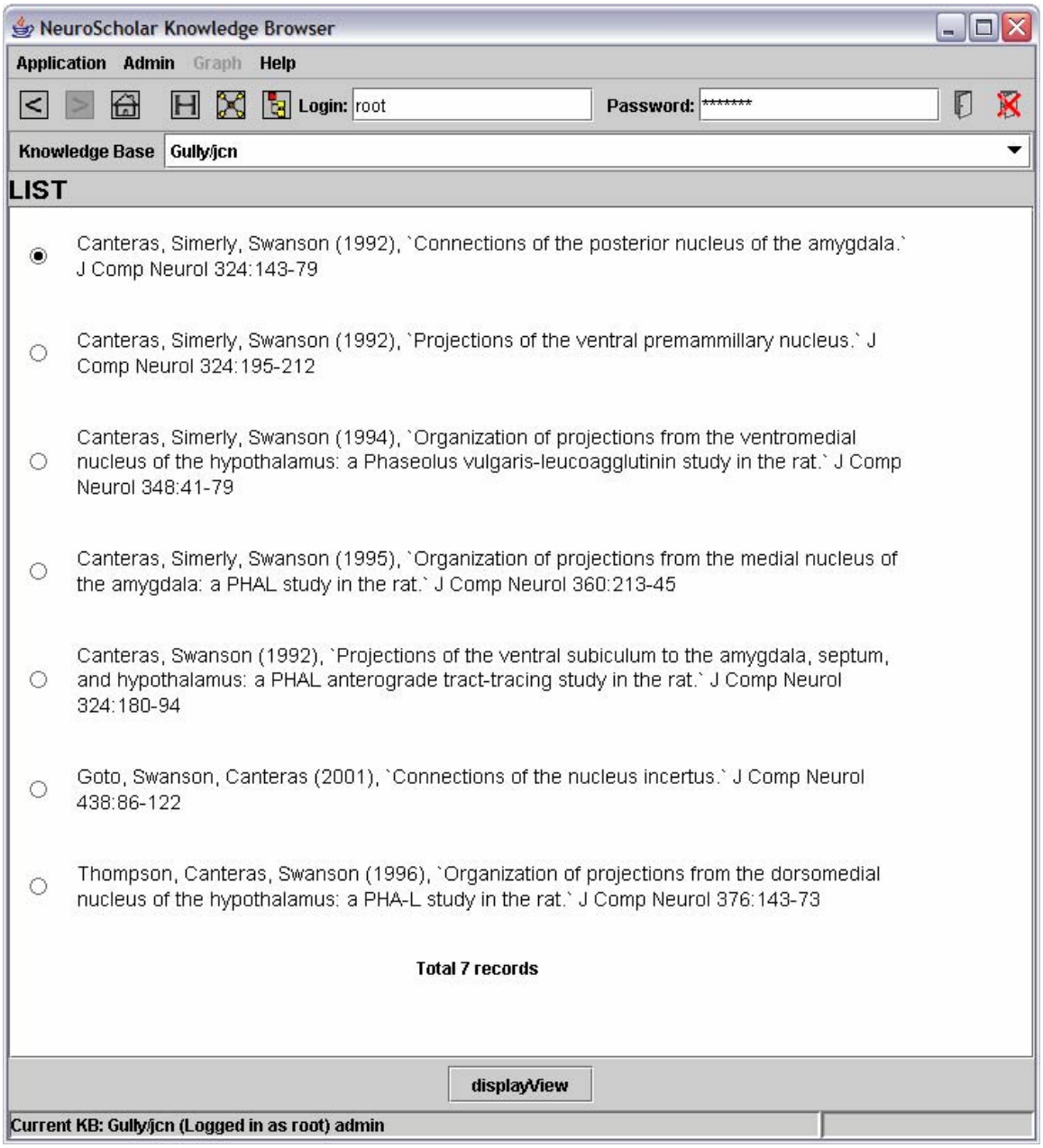
that the system will search for views that satisfy all criteria. Hit the 'executeQuery' button to run the query, hit 'clearQuery' to clear the form.



5.1.4. 'List'

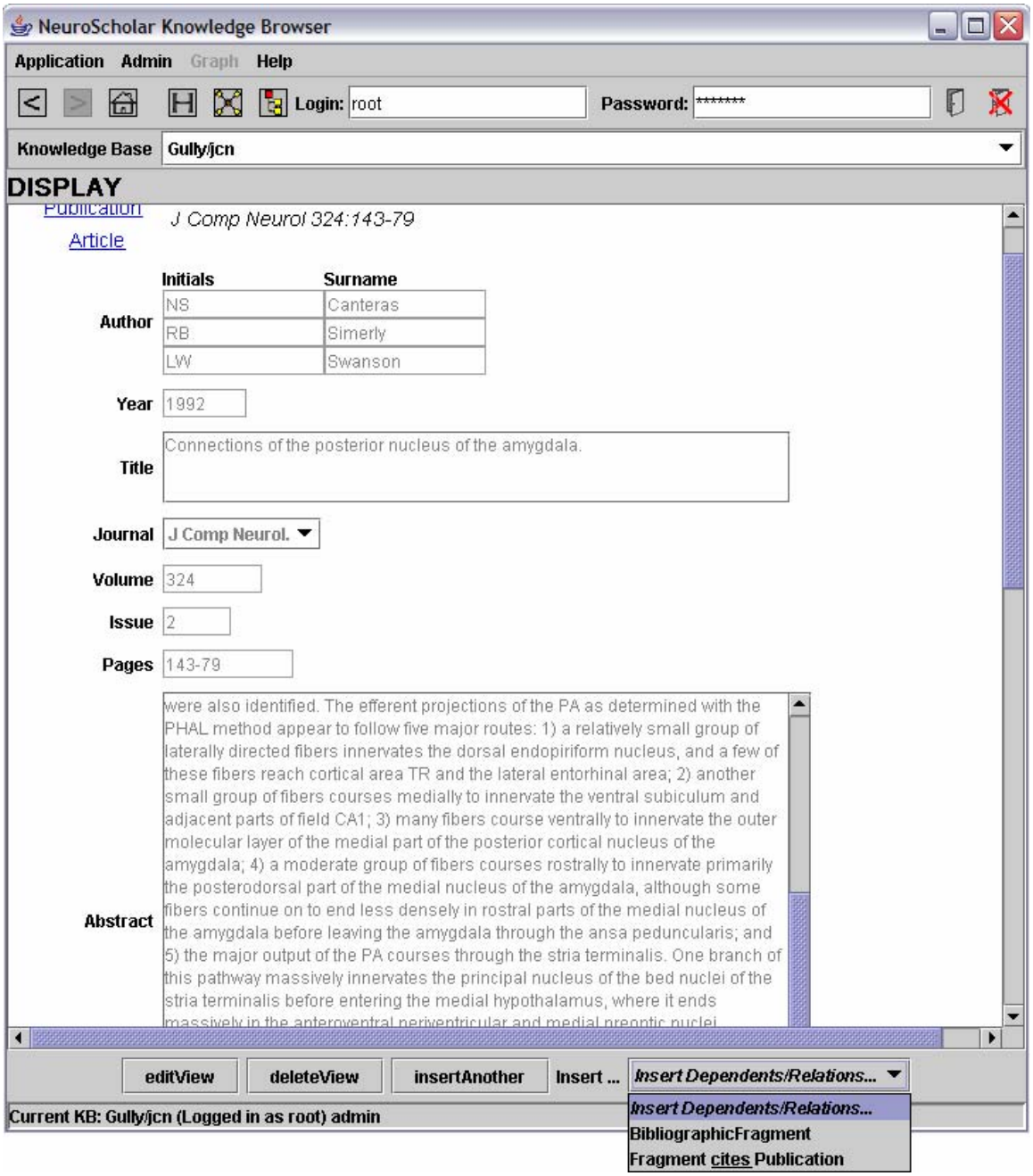
Upon execution of the query, the system will list all entries in the system that satisfy the stipulated criteria. In our example, seven articles were found. These records are listed in order of their indexes (which correspond, in this

case to a citation that you might find in a reference). If more than 100 records are returned, they are split into pages where the user may click a link at the bottom of the list to view the next 100. The user may select a specific view and then hit ‘displayView’ to move to the next state.


NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/jcn
LIST
Canteras, Simerly, Swanson (1992), `Connections of the posterior nucleus of the amygdala.` J Comp Neurol 324:143-79
Canteras, Simerly, Swanson (1992), `Projections of the ventral premammillary nucleus.` J Comp Neurol 324:195-212
Canteras, Simerly, Swanson (1994), `Organization of projections from the ventromedial nucleus of the hypothalamus: a Phaseolus vulgaris-leucoagglutinin study in the rat.` J Comp Neurol 348:41-79
Canteras, Simerly, Swanson (1995), `Organization of projections from the medial nucleus of the amygdala: a PHAL study in the rat.` J Comp Neurol 360:213-45
Canteras, Swanson (1992), `Projections of the ventral subiculum to the amygdala, septum, and hypothalamus: a PHAL anterograde tract-tracing study in the rat.` J Comp Neurol 324:180-94
Goto, Swanson, Canteras (2001), `Connections of the nucleus incertus.` J Comp Neurol 438:86-122
Thompson, Canteras, Swanson (1996), `Organization of projections from the dorsomedial nucleus of the hypothalamus: a PHA-L study in the rat.` J Comp Neurol 376:143-73
Total 7 records
displayView
Current KB: Gully/jcn (Logged in as root) admin

5.1.5. 'Display'

The view, when displayed, appears in a form with the same structure as the query form with the fields filled in. At the top left are links to the views parents (which when clicked provide show the same view in the form of the parent). The index string of the view appears at the top of the form.


NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/jcn
DISPLAY
Publication
Article
J Comp Neurol 324:143-79
Initials
Surname
Author
NS Canteras
RB Simerly
LW Swanson
Year 1992
Title Connections of the posterior nucleus of the amygdala.
Journal J Comp Neurol.
Volume 324
Issue 2
Pages 143-79
Abstract
were also identified. The efferent projections of the PA as determined with the PHAL method appear to follow five major routes: 1) a relatively small group of laterally directed fibers innervates the dorsal endopiriform nucleus, and a few of these fibers reach cortical area TR and the lateral entorhinal area; 2) another small group of fibers courses medially to innervate the ventral subiculum and adjacent parts of field CA1; 3) many fibers course ventrally to innervate the outer molecular layer of the medial part of the posterior cortical nucleus of the amygdala; 4) a moderate group of fibers courses rostrally to innervate primarily the posterodorsal part of the medial nucleus of the amygdala, although some fibers continue on to end less densely in rostral parts of the medial nucleus of the amygdala before leaving the amygdala through the ansa peduncularis; and 5) the major output of the PA courses through the stria terminalis. One branch of this pathway massively innervates the principal nucleus of the bed nuclei of the stria terminalis before entering the medial hypothalamus, where it ends
editView
deleteView
insertAnother
Insert ...
Insert Dependents/Relations...
Insert Dependents/Relations...
BibliographicFragment
Fragment cites Publication
Current KB: Gully/jcn (Logged in as root) admin

Note that individual controls may have quite complex functionality depending on the view type.

If the user is logged in, several buttons appear at the bottom of the view panel. If the user clicks the ‘editVIew’ button, the user will be taken to the edit state (see below). If the user clicks the ‘deleteView’ button, he/she will delete the view along with any other views that are dependent on it before being returned to the ‘documentation’ state for this view definition. If the user clicks the ‘insertAnother’ button, he/she will be taken to a blank ‘insert’ form for this view (see below). A pull-down box permits the user to insert dependent views (such as a Bibliographic Fragment from an Article) or relations (such as a ‘*cites*’ relation).

5.1.6. ‘Insert’

If the user is logged in, he/she may insert a new view. In this case, the user is presented with a blank form, where he/she may enter data and when completed, hit the ‘commitInsert’ button at the bottom of the view panel to finalize the insertion into the database.

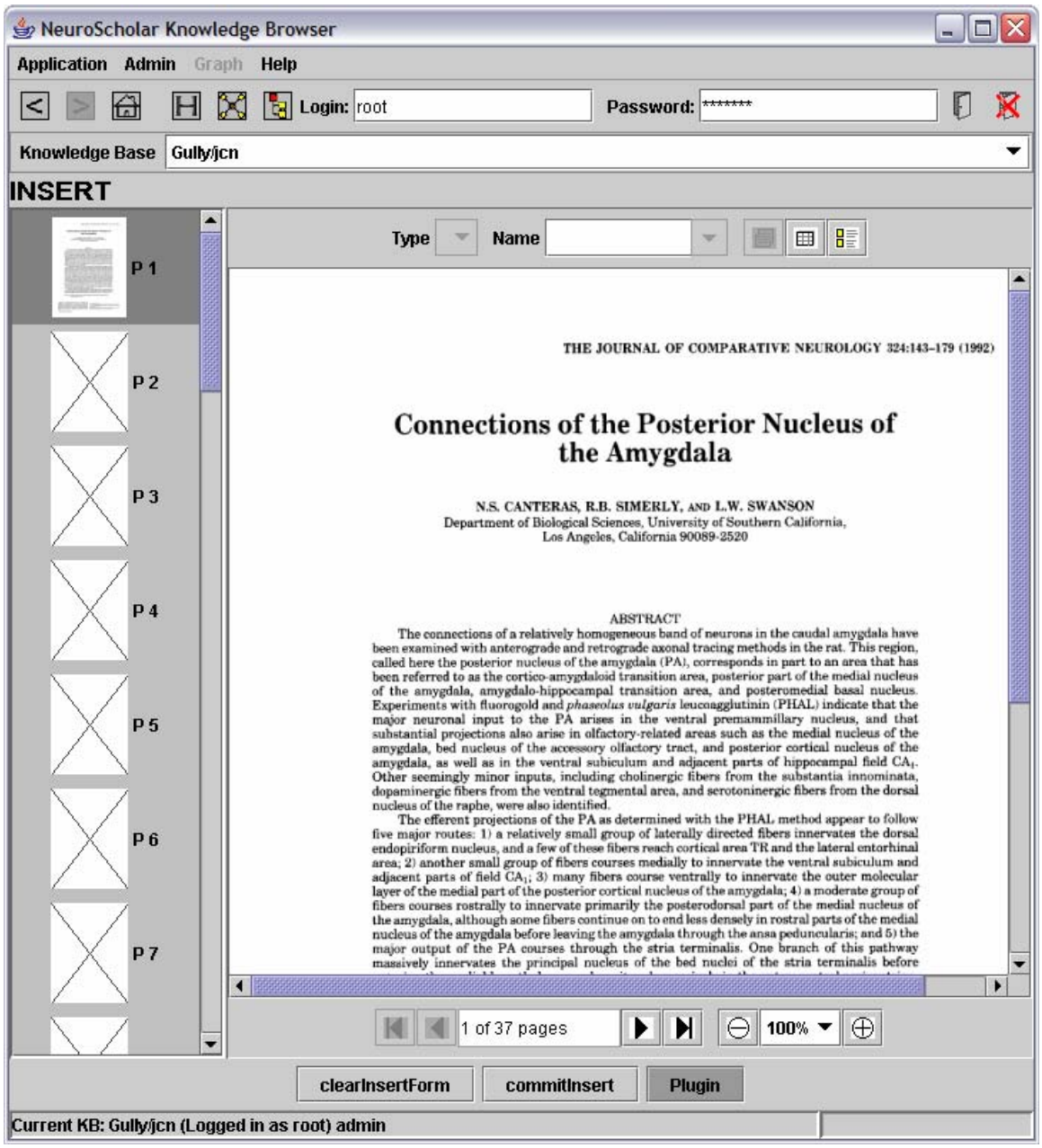
5.1.7. ‘Edit’

If the user is logged in, he/she may edit existing views. In this case, the user is presented with the same form as shown in the display view, but all fields become active and editable. The user simply makes changes as required and then hits ‘commitEdit’ to upload those changes to the knowledge base. If the user wishes not to commit those changes, hitting ‘cancelEdit’ will return them to the previous display state.

5.2. Plugins

Specialized plugins, such as the Fragmenter (shown below) permit the users far more complex control over their data than a simple textual form. However, if the user wishes to view the data in that form, the ‘Plugin’ toggle button permits them

to flip between the text-based data (as shown in the previous section) and the more elegant 'plugin' view (shown below).


NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/jcn
INSERT
Type
Name
P 1
P 2
P 3
P 4
P 5
P 6
P 7
THE JOURNAL OF COMPARATIVE NEUROLOGY 324:143-179 (1992)
Connections of the Posterior Nucleus of the Amygdala
N.S. CANTERAS, R.B. SIMERLY, AND L.W. SWANSON
Department of Biological Sciences, University of Southern California, Los Angeles, California 90089-2520
ABSTRACT
The connections of a relatively homogeneous band of neurons in the caudal amygdala have been examined with anterograde and retrograde axonal tracing methods in the rat. This region, called here the posterior nucleus of the amygdala (PA), corresponds in part to an area that has been referred to as the cortico-amygdaloid transition area, posterior part of the medial nucleus of the amygdala, amygdalo-hippocampal transition area, and posteromedial basal nucleus. Experiments with fluorogold and phaseolus vulgaris leucoagglutinin (PHAL) indicate that the major neuronal input to the PA arises in the ventral premammillary nucleus, and that substantial projections also arise in olfactory-related areas such as the medial nucleus of the amygdala, bed nucleus of the accessory olfactory tract, and posterior cortical nucleus of the amygdala, as well as in the ventral subiculum and adjacent parts of hippocampal field CA1. Other seemingly minor inputs, including cholinergic fibers from the substantia innominata, dopaminergic fibers from the ventral tegmental area, and serotoninergic fibers from the dorsal nucleus of the raphe, were also identified.
The efferent projections of the PA as determined with the PHAL method appear to follow five major routes: 1) a relatively small group of laterally directed fibers innervates the dorsal endopiriform nucleus, and a few of these fibers reach cortical area TR and the lateral entorhinal area; 2) another small group of fibers courses medially to innervate the ventral subiculum and adjacent parts of field CA1; 3) many fibers course ventrally to innervate the outer molecular layer of the medial part of the posterior cortical nucleus of the amygdala; 4) a moderate group of fibers courses rostrally to innervate primarily the posterodorsal part of the medial nucleus of the amygdala, although some fibers continue on to end less densely in rostral parts of the medial nucleus of the amygdala before leaving the amygdala through the ansa peduncularis; and 5) the major output of the PA courses through the stria terminalis. One branch of this pathway massively innervates the principal nucleus of the bed nuclei of the stria terminalis before
1 of 37 pages
100%
clearInsertForm
commitInsert
Plugin
Current KB: Gully/jcn (Logged in as root) admin

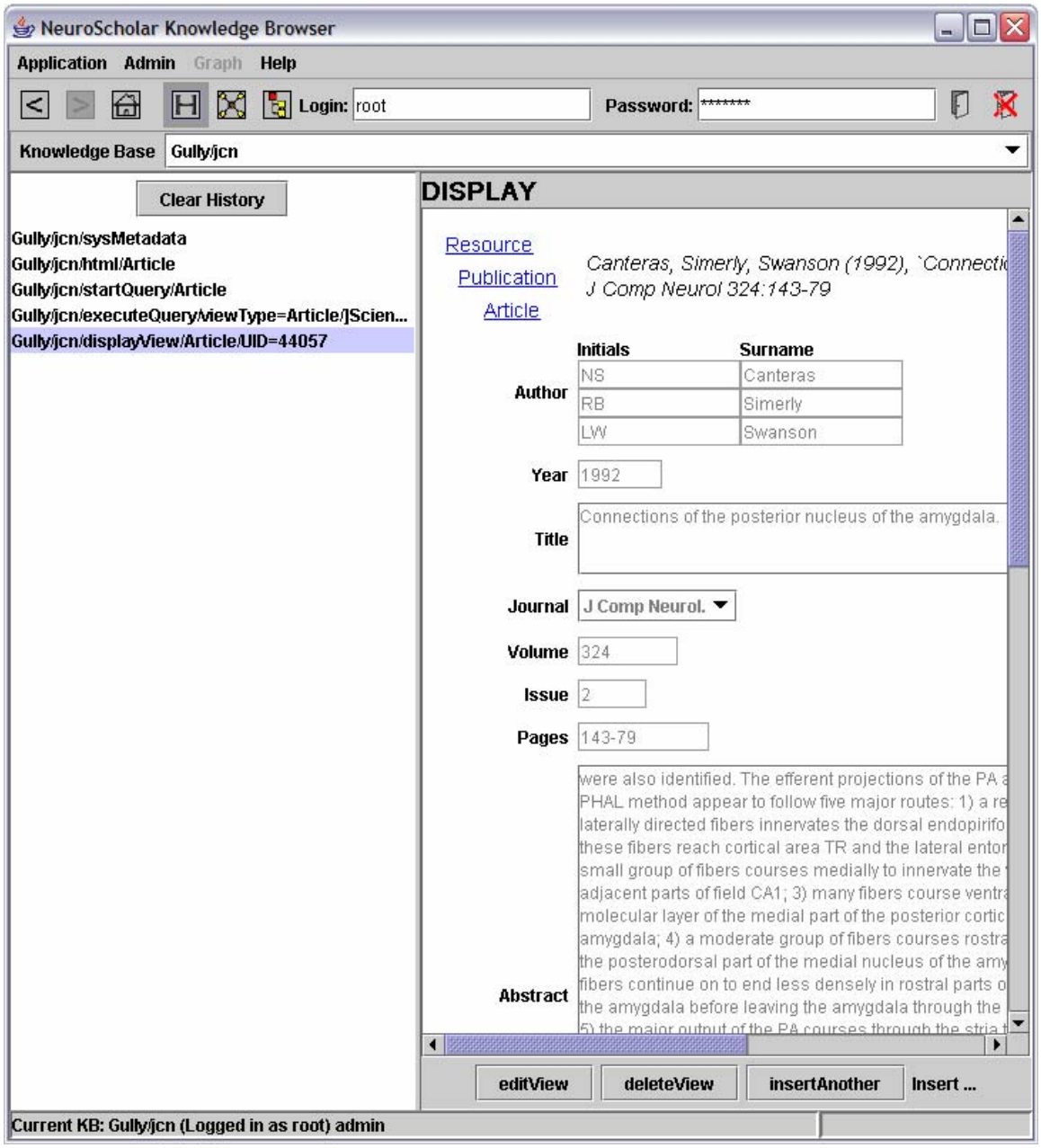
5.3. 'Knowledge Browser' Toolbar commands

The vast majority of computer users today are web-users and we have designed the NeuroScholar user interface to provide basic 'web-like' navigational capabilities. The toolbar has a back button, a forward button and a button for 'home'. This permits users to go back to previous states of the system and then

advance forward to subsequent states. The home button resets the current KB and returns them to the ‘home’ state displaying counts of all views in the system.

5.4. ‘History’ commands

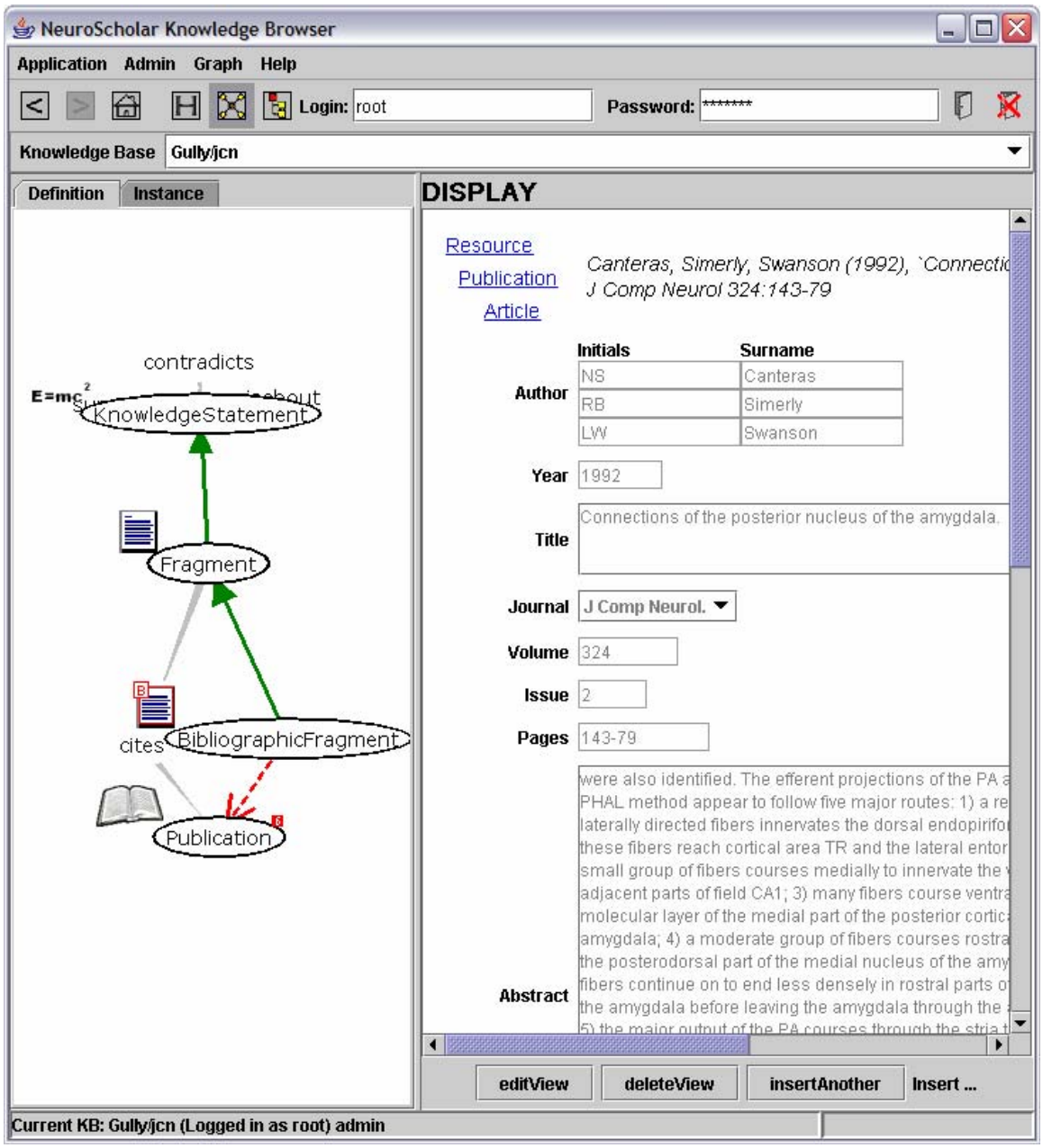
The toolbar also contains a toggle button that permits the user to see their ‘history’. This is shown below. The user may click on any of the history items listed to return to that item. If the user then does something in the system *other than* perform another history-based action, then that action will become the next item in the history list and all subsequent history items will be removed. The history pane can be removed by clicking on the History toggle button a second time.



5.5. ‘Graph’ commands

NeuroScholar is based on the design of a network of nodes and edges (a ‘graph’) and it is possible to view both the definition of the data model and the data itself according to that design. To activate the graph pane, click the ‘graph’ button in the tool bar:

This also activates the ‘Graph’ menu.



The Graph panel has two tabs: Definition and Instance.

5.5.1. The Graph Panel – Definition Pane

This pane shows the design of the knowledge base that is currently loaded. It may be quite complex and is based on a visual notation called the 'Unified Modeling Language' which is widely used in the computer industrial to describe data schemata. Essentially the design shows the organization of view definitions

in the system and their possible interrelationships. You may interact with this representation by left- or right- clicking on the nodes, the edges or the background.

Note: Mac OS X users should use shift-click where right-click is mentioned in the text (i.e., hold the shift key down whilst clicking with the mouse).

The screen shot shown above illustrates the main concepts required to understand the components of the graph shown in the View Definition panel.

- Each node is a separate view. Data views have oval borders with an accompanying icon whereas relation views have no borders.
- Children are connected to their parents by green, ‘inheritence’ arrows.
- Simple constitutive links between views that are blue tapering arrows (not shown in screenshot)
- Dependent views are connected to the views they depend on by red, dashed arrows.
- Relations have one input and one output (but they have the same view for both).

Double clicking on a node brings it to the center and opens all links to it.

Right-clicking a node brings a number of options depending on the type of node and whether the user is logged in.

If the user has not logged in, all options available here are simply display functions. The user may view documentation (which takes them to the ‘Documentation’ state this view). The user may also run a query for the specific type of view. Note that the Expand/Collapse/Hide/Center Node functions all have a superficial effect on the view graph by permitting the user to remove specific items from view temporarily. This is not the same as the system’s visibility controls (which will be described presently). Note that if nodes have been loaded into the application’s memory but are temporarily hidden, a small red box will

appear on the top right edge of the node with the number of neighboring hidden nodes.

If the user has logged in, the user may also insert an instance of the view. Note that this option does not appear if the view being clicked has dependencies (e.g., the *BibliographicFragment* view definitions and its children are dependent on the *Article* view).

If the user selects ‘Importance…’ from the popup menu, a small box with five stars appears, displaying the relative importance of this view definition when prioritizing views for reliability (note that this feature is still under development).

Note that, in this case, relation nodes have identical behavior except that they do not permit users to enter or view ‘Importance…’ data.

It is possible to temporarily remove an edge from the graph by right clicking it and selecting ‘Hide Edge’ from the popup menu.

If a user right-clicks the background of the viewgraph definition panel a ‘Toggle Controls’ popup menu will appear. If this is selected the controls shown in the figure appear. These controls are generic to the TouchGraph interface and can be manipulated to control the appearance of the overall graph.

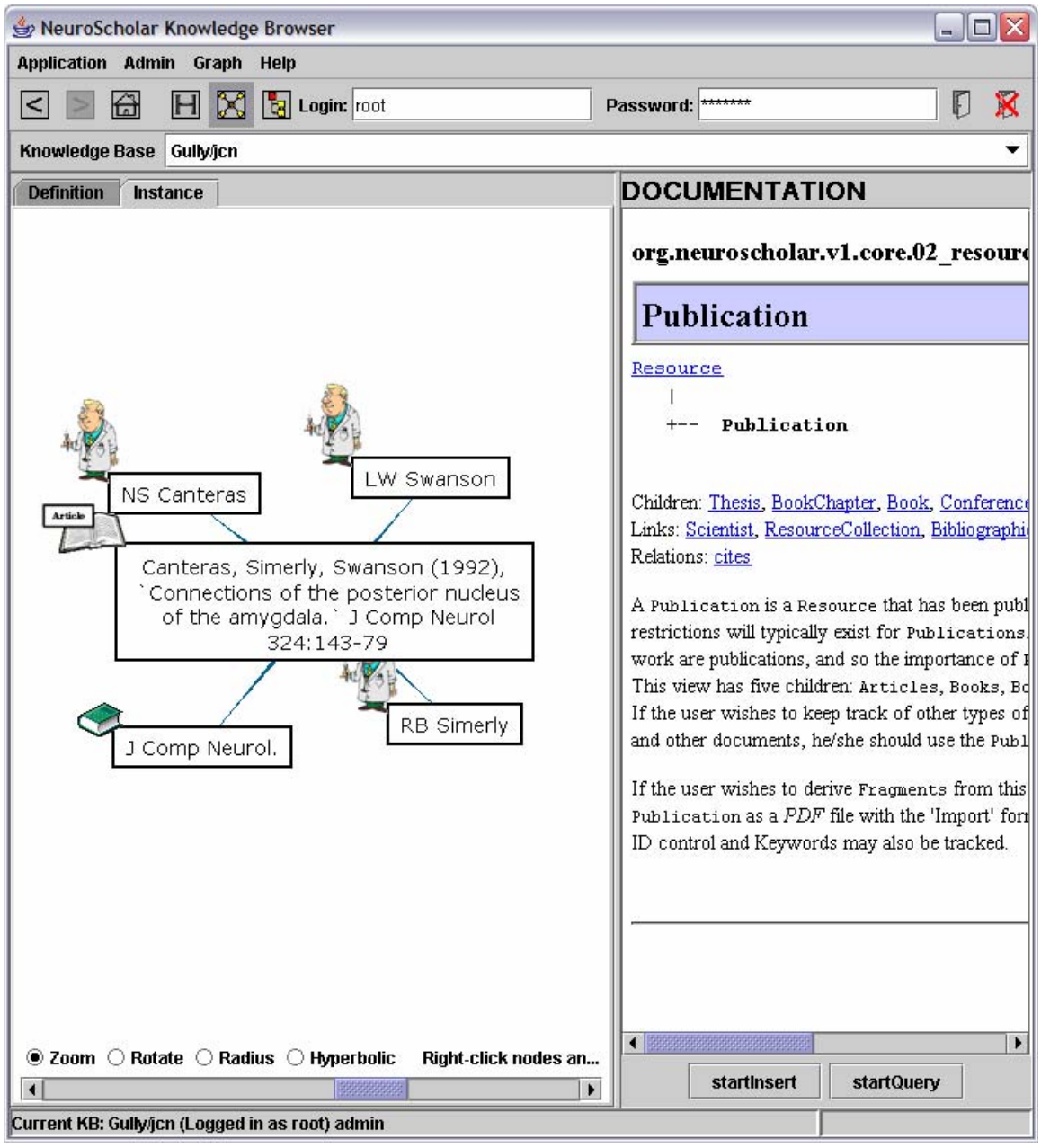
The **Zoom** control scales distances within the graph so that nodes lies closer together or further apart.

The **Rotate** control spins the whole graph.

The **Radius** control allows you to manipulate the extent of the graph that appears relative to the central node of the graph. You can expand the radius to show every node, or you can shrink it to show only the central node with none of its immediate neighbors.

The **Hyperbolic** control permits the extent of apparent lensing within the panel. This causes nodes in center of the panel to lie further apart than nodes at the panels’ edges.

5.5.2. The Graph Panel – Instance Pane


NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/jcn
Definition Instance
DOCUMENTATION
org.neuroscholar.v1.core.02_resourc
Publication
Resource
|
+-- Publication
NS Canteras
LW Swanson
Canteras, Simerly, Swanson (1992), `Connections of the posterior nucleus of the amygdala.` J Comp Neurol 324:143-79
RB Simerly
J Comp Neurol.
Children: Thesis, BookChapter, Book, Conferenc
Links: Scientist, ResourceCollection, Bibliographi
Relations: cites
A Publication is a Resource that has been publ
restrictions will typically exist for Publications.
work are publications, and so the importance of
This view has five children: Articles, Books, Bo
If the user wishes to keep track of other types of
and other documents, he/she should use the Publ
If the user wishes to derive Fragments from this
Publication as a PDF file with the 'Import' for
ID control and Keywords may also be tracked.
Zoom Rotate Radius Hyperbolic Right-click nodes an...
startInsert
startQuery
Current KB: Gully/jcn (Logged in as root) admin

Whenever a view instance is loaded in the Display state in the view panel, a node corresponding to it is loaded in the instance pane. All of the node's visible neighbors will also be displayed in the pane. The act of double-clicking a node will bring it to the center of the graph and load all of its neighbors, links and relations and will also load it for viewing in the right-hand panel. Thus it is easy to

navigate the contents of the knowledge base by simply double-clicking nodes and traversing the relations that appear.

As with the viewgraph definition pane, the user may interact with the graph by right clicking nodes, edges and the background. Right-clicking the background produces exactly the same effect here as in the viewgraph definition pane, and will not be covered again here.

View instance nodes appear as shown in the screen-shot above. This depicts an *Article* view instance where the node's label is longer than one line and so is truncated until the user passes their mouse over the label, where it expands to display itself fully.

As before the functions available to users from right-clicking nodes depends on the type of node and the login status of the user.

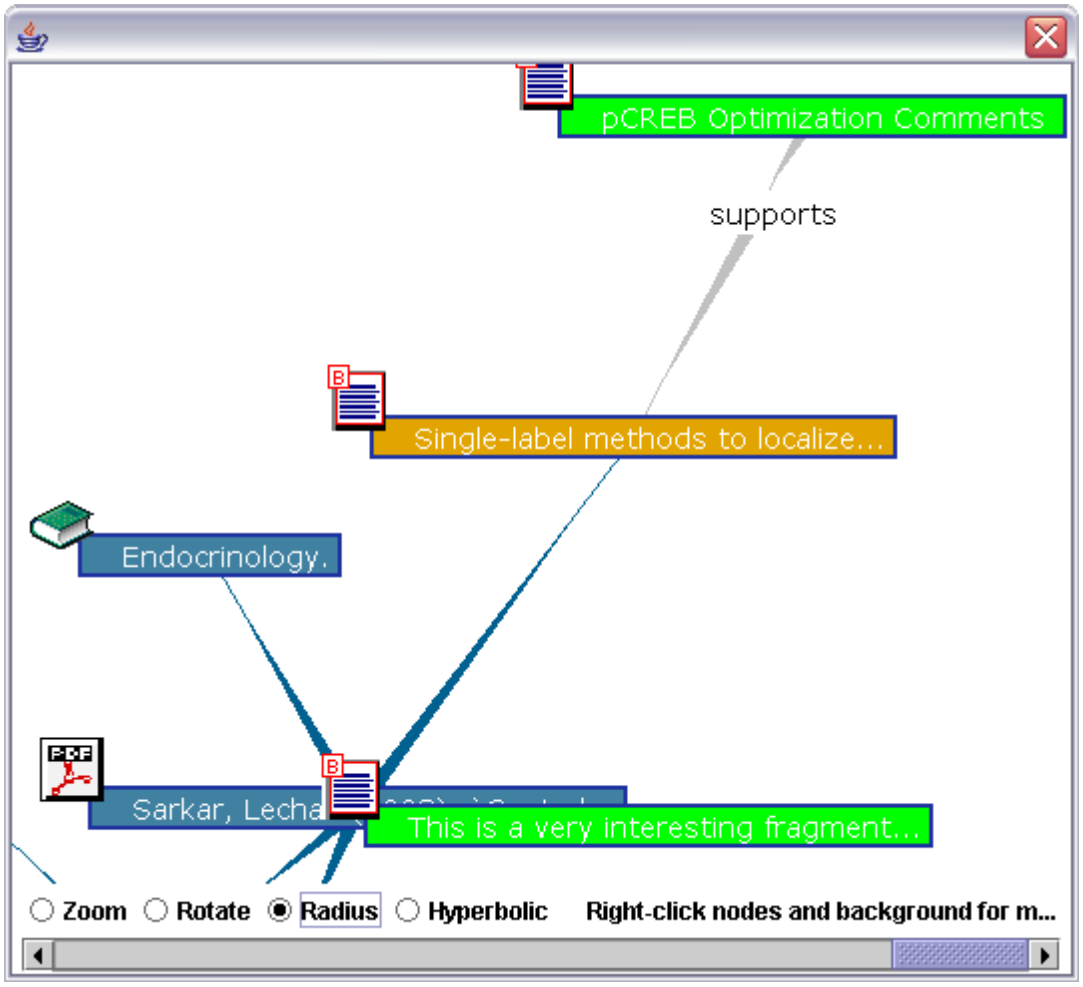
As before, if the user has not logged in, all options available here are simply display functions. The user may view the contents of the view itself by selecting 'Display this View Instance' or he/she may list any views linked or related to the clicked view by selecting 'List Linked view instance >'. In this case another menu of all the possible types of view pops up so that the user may select the required list. This executes a list command on the system and displays the data in the right-hand panel.

If the user has logged in, a far larger number of options are available. The user may insert an instance of a specified view definition (thus permitting *BibliographicFragment* instances to be inserted from a given *Article* instance).

If the user selects 'Opinion…' from the popup menu, a small box with five stars appears, displaying the user's opinion of how reliable the data represented in this instance is.

On clicking the 'Insert Relation…' menu item, the system will pop up a list of possible relations that may be inserted. If the user selects one of these, the system will pop up a new frame with a version of the view instance graph that is

currently in memory, currently centered on the node being right-clicked on. Each node in the graph that is eligible to be the recipient of the relation appears in green. After identifying the target for the relation, right click the target and select 'Insert Relationship' from the pop up menu. The system will then insert the appropriate relation into the graph. This feature is somewhat cumbersome and may be dropped from future versions of the system.

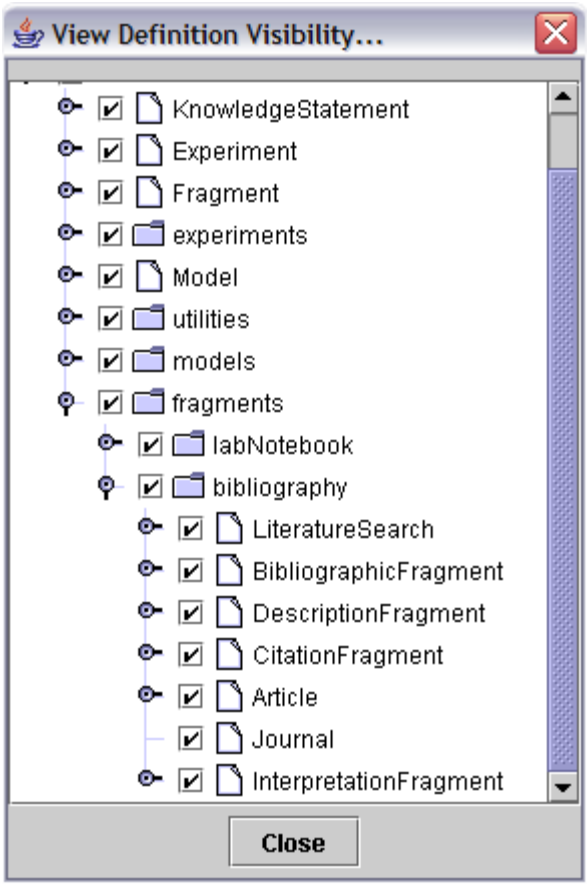


Relation nodes behave in the same way that data nodes do when the user is not logged in. When the user is logged in however, the user may delete the relation from the viewgraph instance pane.

5.5.3. Visibility Controls

Nodes may be made temporarily invisible through the controls on the viewgraph panes. It may be required by the user to manipulate the visibility of nodes more directly and by selecting ‘Graph > View Visibility… > View Definition Visibility…’ or ‘Graph > View Visibility… > View Instance Visibility…’, the user may load or dismiss views from the view graph currently loaded in memory.

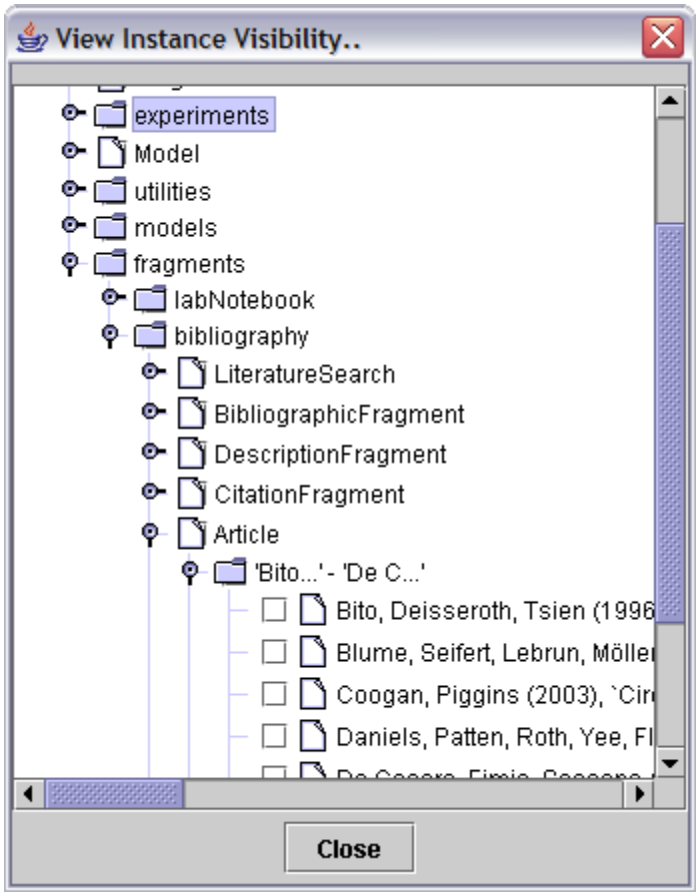
5.5.3.1. View Definition Visibility



When the View Definition Visibility control is selected, a tree with checkboxes on each node opens up. The tree corresponds to the hierarchical package design of the UML model that underlies the system (and so, a good understanding of the data model being used in the system is required). In the tree, each folder contains a number of views (and if the user unchecks a folder, all contained views within the folder will be unchecked as well). Each document icon within the

tree corresponds to a View Definition and if these are unchecked they effectively disappear from the data model (i.e., they are removed from the viewgraph definition pane and any of their instances are removed from the viewgraph instance pane).

5.5.3.2. View Instance Visibility



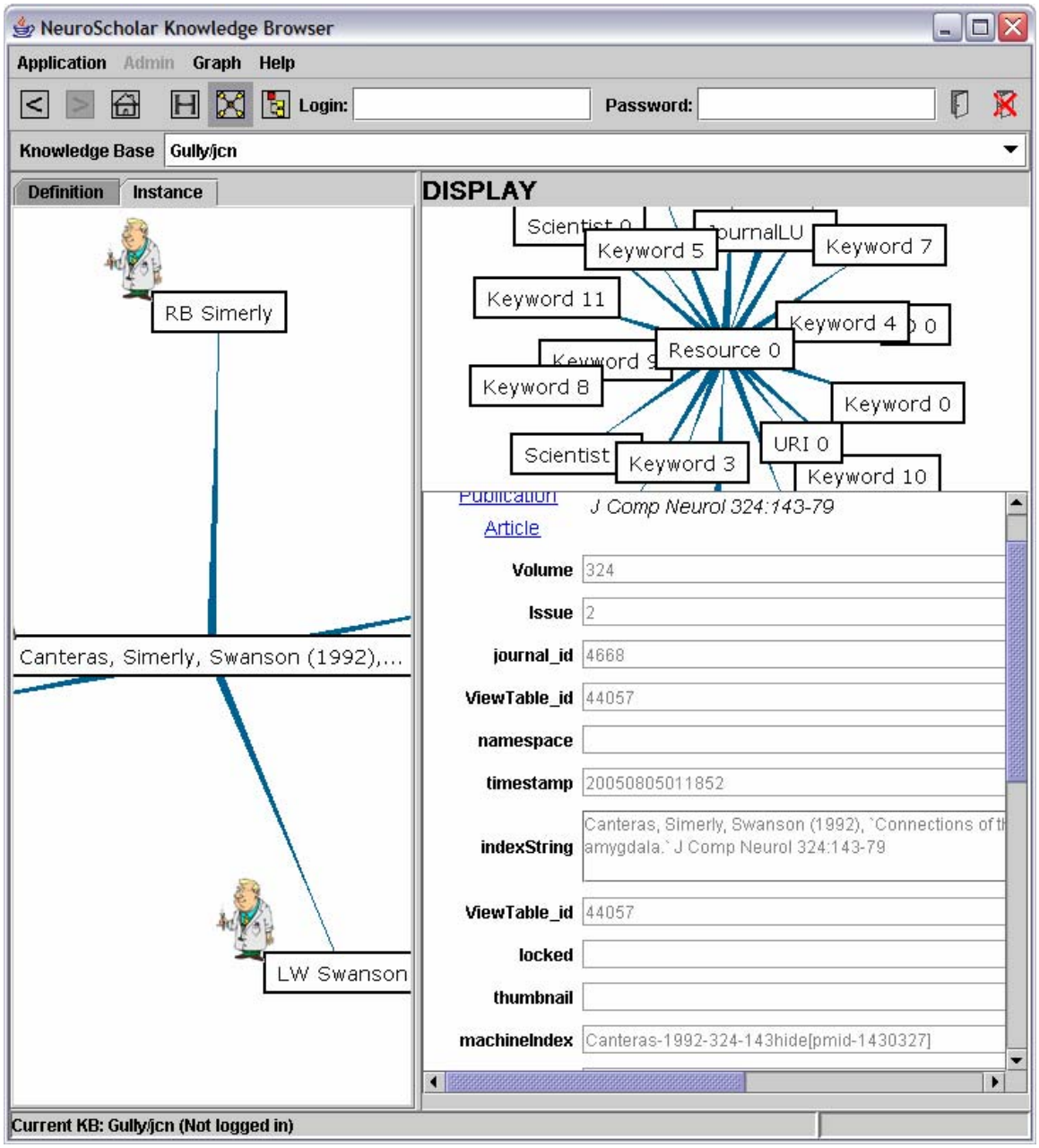
Similar to the View Definition Visibility control, the View Instance Visibility control opens a tree based on the hierarchical package design of the UML model that underlies the system. In this tree, the folders and views do not have checkboxes. The user may open up a specific view and alphabetically browse the contents of the knowledge base. Each view instance will have checkbox, which will load it (and its immediate neighbors) into the view instance panel. This is a useful feature to permit the user to load specific views into the viewgraph instance pane quickly.

5.5.3.3. Profiles

These are predefined visibility controls for a collection of view definitions. Select ‘Graph > Profiles >’ and a list of profiles will come up. With a large, complex model, this permits the user to concentrate only on a specific subtask.

5.5.4. An aside: the substructure of a view

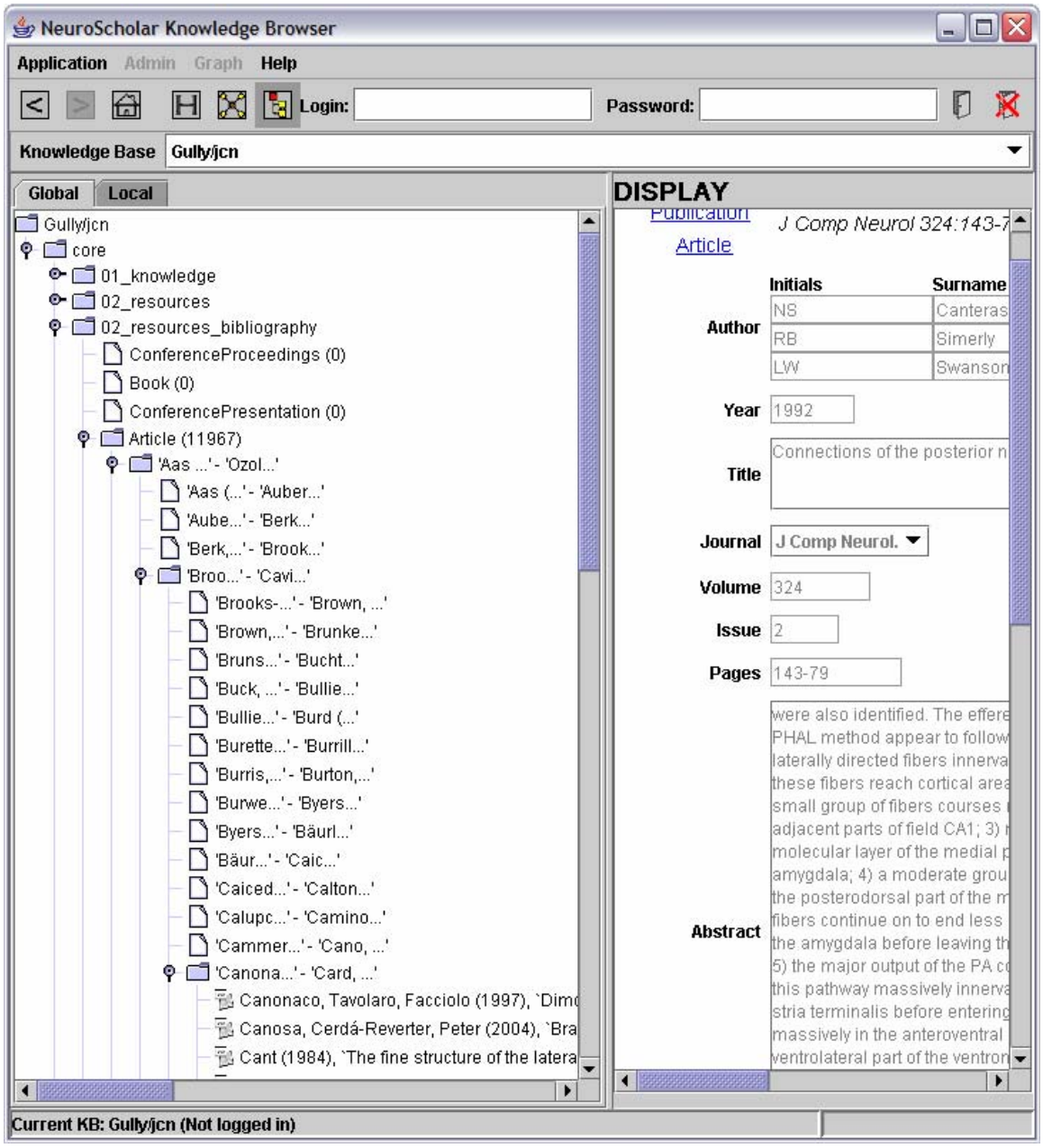
Each view has a graph-like substructure that may be accessible for viewing by selecting ‘Views > Show View Substructure …’. The view panel shows all primitives for the current view with all data in a generic form. It is possible to look at any of the data in the other primitives simply by right clicking the primitive and selecting ‘Show this primitive’ from the popup menu.



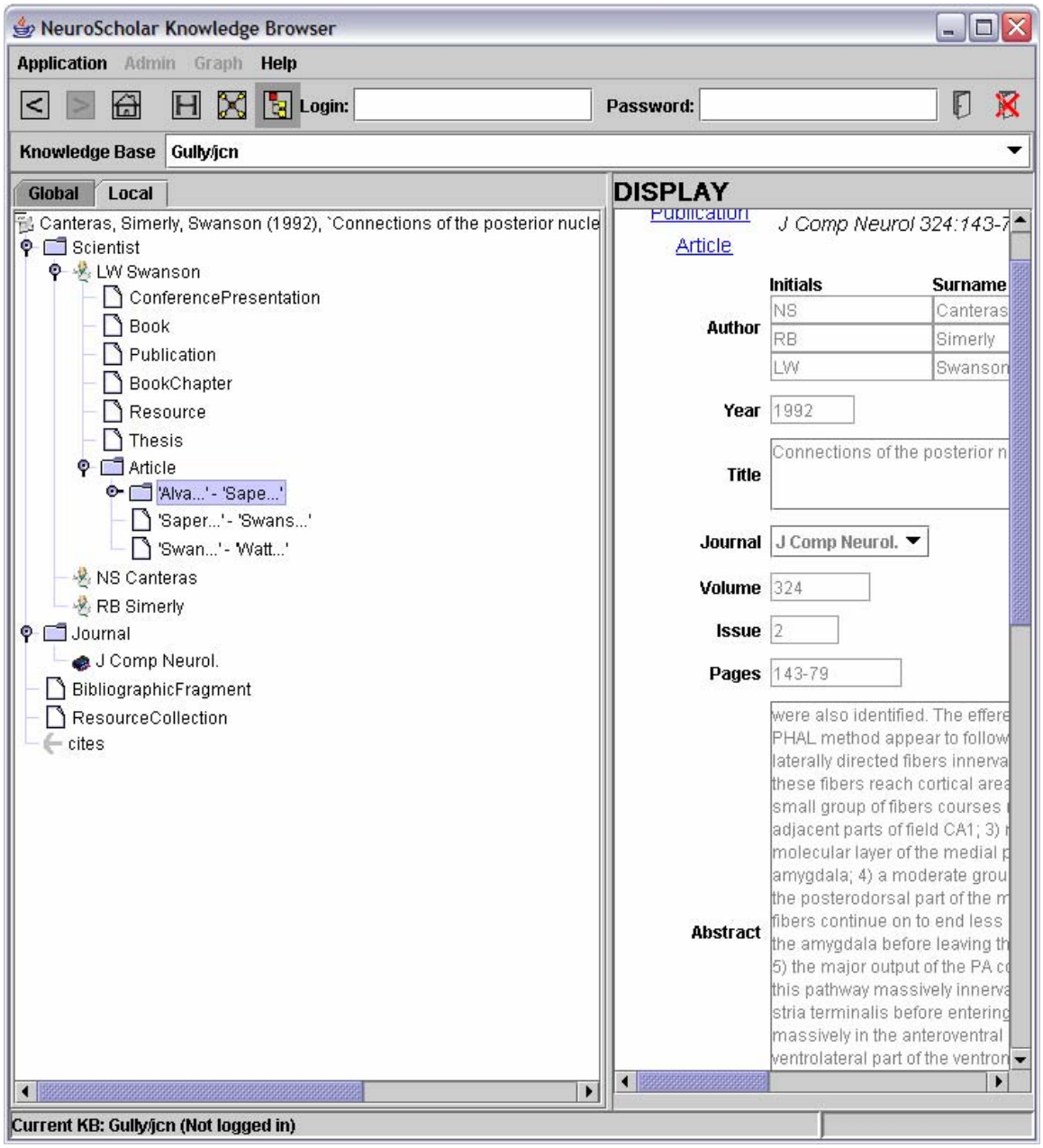
5.6. ‘Tree’ commands

The user may view the contents of the KB as a tree by selecting the Tree toggle button from the Toolbar shown as:

This opens the tree panel on the left hand side. This panel has two tabs: ‘global’ and ‘local’. The global tab is simply a shallow tree representing the entire contents of the knowledge base organized by package and view type (shown below).



The local tab permits the user to browse the immediate neighborhood of a view by first dragging and dropping the current view from the right hand panel to the left. This opens up a display where the user may click on folders representing that view's links and relations to other items in the knowledge base. The use may then open those items to view. In the following example, the articles of one of the authors of the paper have been expanded to be viewed.

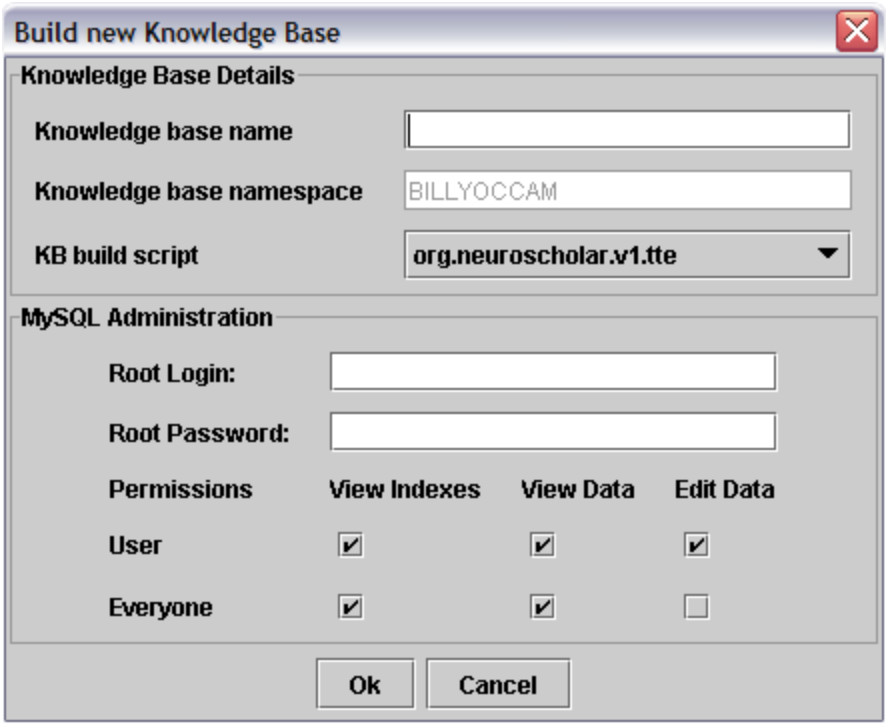


6. System Administration How-tos

The NeuroScholar system is built on an underlying MySQL database and the user may manipulate the database through MySQL. This involves the creation, destruction, saving and restoration of knowledge bases. To perform these functions, the user must login as root on the browser itself to activate the 'Admin' menu.

6.1. Creating a new knowledge base

1. To create a new knowledge base select 'Admin > Create KB'. The following dialog pox pops up:

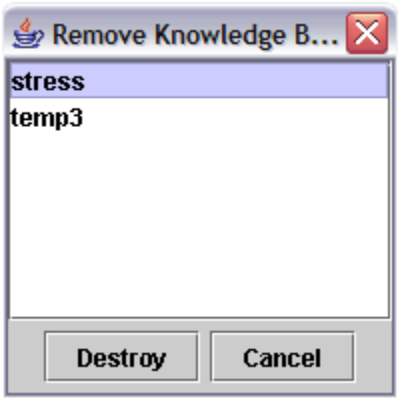


The system automatically fills in the details of the knowledge base namespace (which is the name of the computer being used). The user is required to fill in the Knowledge Base name and select the type of KB they desire (within this release, only the 'www.neuroscholar.v1.core' is available).

2. Fill in the root login and password. The user may also set the permissions for this knowledge base restricting who may or may not access or edit the data in the system.
3. Hit ‘Ok’. A small progress monitor dialog will pop up and when this is complete, you should be in the ‘Home’ state for an empty KB with the correct name and design.

6.2. Destroying a knowledge base

1. Select ‘Admin > Destroy KB’. The following dialog pox pops up:



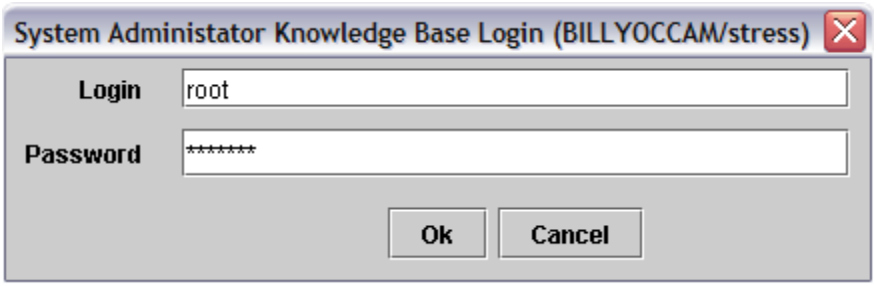
2. Select one of the listed systems, then hit the ‘Destroy’ button.
3. Confirm your root status by logging in as root.
4. Then the system will drop the relevant database from within the underlying MySQL system. Note that this ***permanently deletes*** all data in that database.
5. The dialog will remain open. So you may repeat the process as necessary.

6.3. Saving a knowledge base

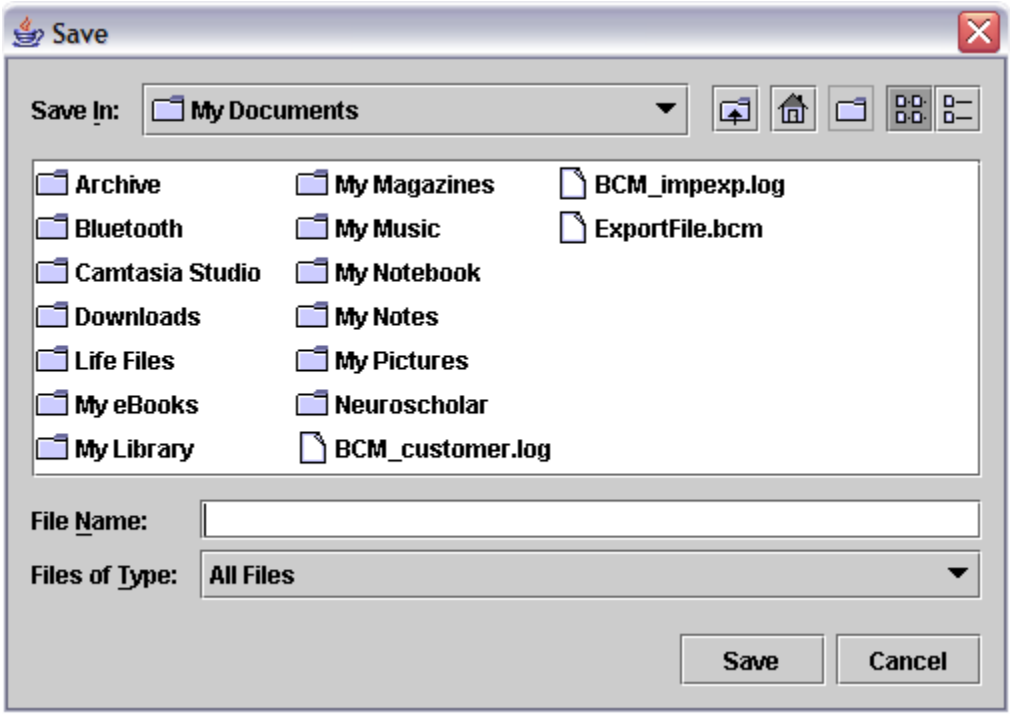
Saving an existing knowledge base to a *.nar (‘NeuroScholar Archive’) file is a particularly useful activity permitting data migration between systems and even knowledge sharing between people.

1. Select ‘Admin > Save Knowledge Base’.

2. Use the login dialog box to confirm your ‘root’ status on the system.



3. A ‘SaveAs…’ dialog box appears. Enter the name and location of where you would like to put your `*.nar` file.

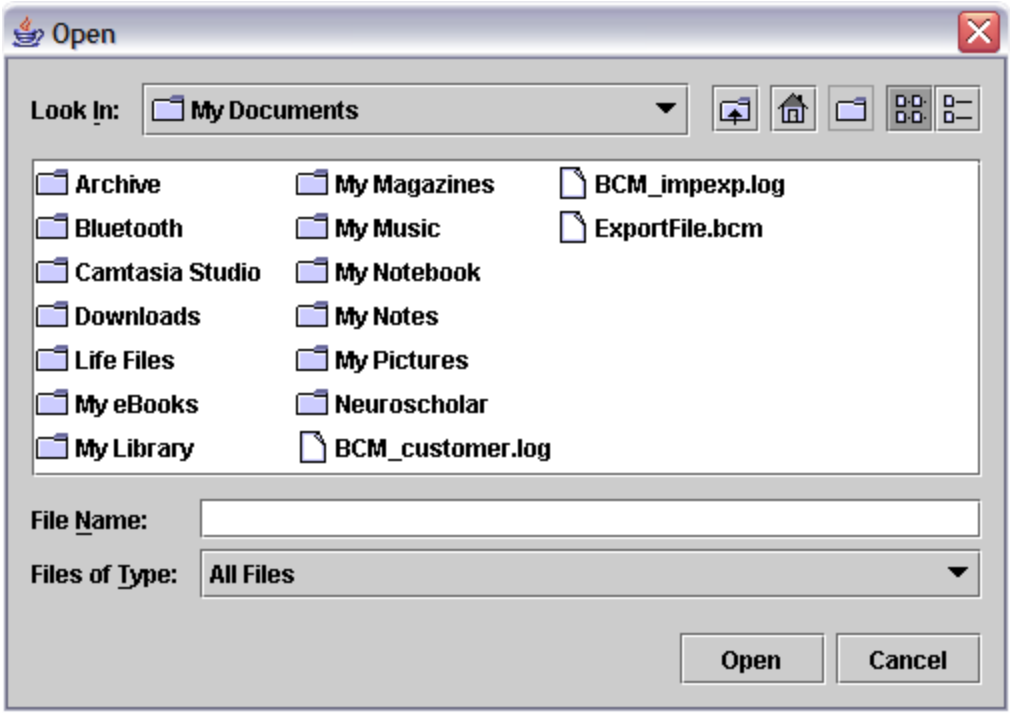


4. When ready, hit ‘Save’.
5. NeuroScholar will dump the entire contents of the current knowledge base to disk as that file.

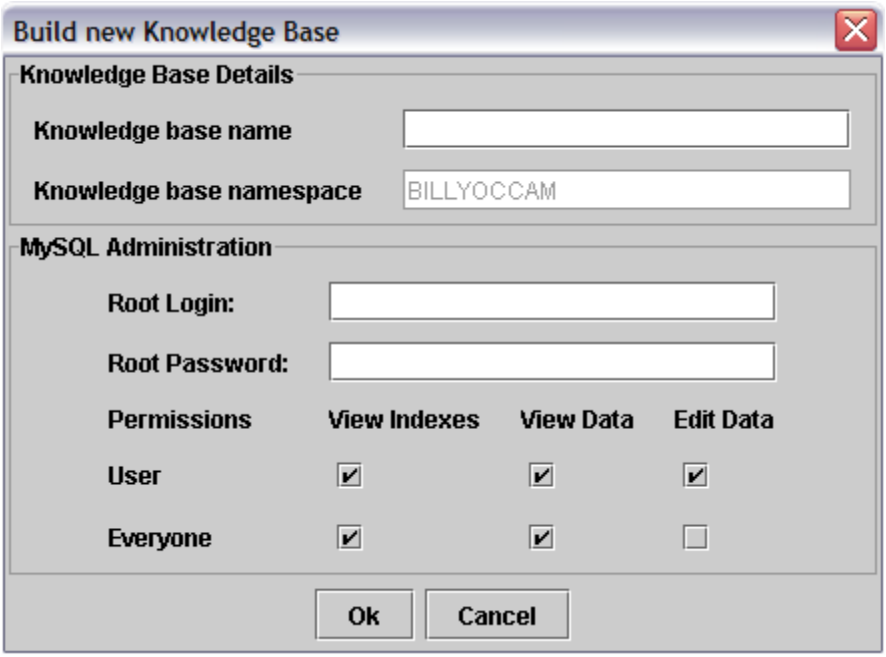
6.4. Restoring a saved knowledge base

After saving the data in a knowledge base to a file, you might want to restore that knowledge to another knowledge base.

1. Select ‘Admin > Restore Knowledge Base from Archive…’. An ‘Open’ dialog box pops up:



2. Select the *.nar file that they wish to restore to the current database and then hit ‘Open’.
3. The system will then pop up a ‘Build new Knowledge Base’ dialog box that does not permit the user to define the system’s ontology (since that is set within the archive). Enter the name and confirm your root status by entering the root login and password onto this form.



Select appropriate permissions for the system and hit ‘OK’.

A small progress monitor dialog will pop up and when this is complete, you should be in the ‘Home’ state for an your restored KB with the name you entered in the previous step.

6.5. Managing Users

Once a knowledge base has been built and selected (note that the logs contain a complete description of all SQL commands performed on the underlying database), it is now possible to define users and knowledge models and then enter real data into the system. Note that this step may be omitted with a knowledge base that has been restored from an archive, since that system will already have well-established preliminary data.

To insert a new user, select ‘Admin > Add New User…’ from the menu. This loads a form where you can insert data describing a new user in the usual way. Note that the initial text of the password is masked by ten characters. Simply fill in the form, and hit ‘commitInsert’. Once that is complete, you may login as this user from the browser.

To list (display and edit) the users in the system, select ‘Admin > List Users…” from the menu. This lists all the users registered for this knowledge base in the system. You may select one to display it and then edit their details if required.

7. The ‘core’ system

The core system of NeuroScholar handles the lowest level of functionality: downloading articles and other bibliographic resources whilst permitting the user to build a repository of knowledge statements based on those resources.

7.1. System packages

The system has a modular design based on series of ‘packages’. This design allows us to construct versatile knowledge bases tailored to different knowledge management tasks, but for now, we restrict our attention to the simplest aspects of managing information in the published literature. Basically the views have been organized according to the following packages

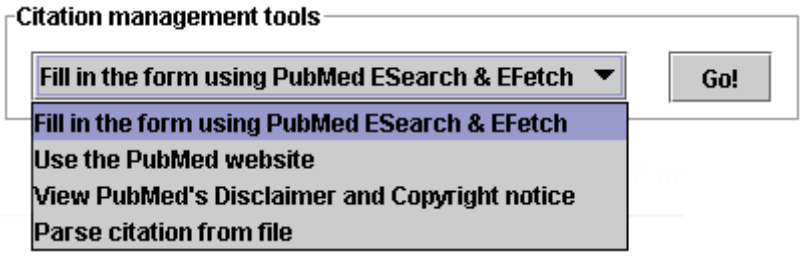
- Knowledge: views in this package are concerned with statements of Knowledge and their relationships.
- Resources: views in this package are concerned with external items that contain knowledge. In this system these are publications, which might be articles, books etc.
- Fragments: views in this package are concerned with excerpts taken from the resources each of which forms a small specialized knowledge statement called a fragment. These fragments are literally annotated excerpts taken from PDF documents within the users’ document library.

8. How-tos

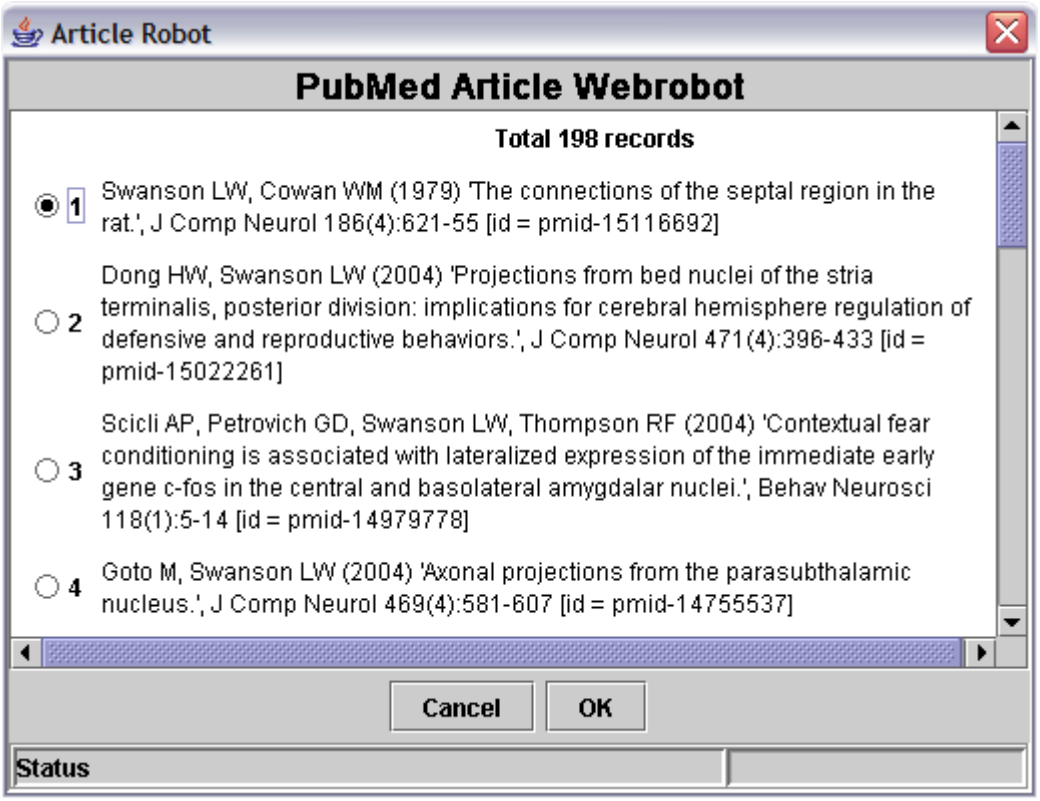
To make this section oriented towards helping users accomplish specific tasks, we have listed a number of unitary tasks with a complete description of how to accomplish that task.

8.1. Enter a bibliographic reference into the system using PubMed

1. Load the required knowledge base
2. Login to the knowledge base (either as root, or as a user you have already created in the system).
3. Select ‘Article’ from the list of Views to view the Documentation for Article views.
4. Click the ‘startInsert’ button
5. If you know the citation details of the paper you are interested in, enter a few details of the reference into the form. The best approach is to enter the last name of the first author, the year, the volume and the page of the start of the article. This always returns a unique reference. If you do not know these details, enter what you do know and continue.
6. Use the ‘Citation management tools’ form control. This is a simple combo box with four options. Select the first ‘Fill in the form using PubMed ESearch & EFetch’ and hit ‘Go!’.


Citation management tools
Fill in the form using PubMed ESearch & EFetch
Go!
Fill in the form using PubMed ESearch & EFetch
Use the PubMed website
View PubMed's Disclaimer and Copyright notice
Parse citation from file

7. The system will be active for a short while and (if the web-connection of the computer is functional) should return a dialog box of the following form.


Article Robot
PubMed Article Webrobot
Total 198 records
1 Swanson LW, Cowan WM (1979) 'The connections of the septal region in the rat.', J Comp Neurol 186(4):621-55 [id = pmid-15116692]
2 Dong HW, Swanson LW (2004) 'Projections from bed nuclei of the stria terminalis, posterior division: implications for cerebral hemisphere regulation of defensive and reproductive behaviors.', J Comp Neurol 471(4):396-433 [id = pmid-15022261]
3 Scicli AP, Petrovich GD, Swanson LW, Thompson RF (2004) 'Contextual fear conditioning is associated with lateralized expression of the immediate early gene c-fos in the central and basolateral amygdalar nuclei.', Behav Neurosci 118(1):5-14 [id = pmid-14979778]
4 Goto M, Swanson LW (2004) 'Axonal projections from the parasubthalamic nucleus.', J Comp Neurol 469(4):581-607 [id = pmid-14755537]
Cancel
OK
Status

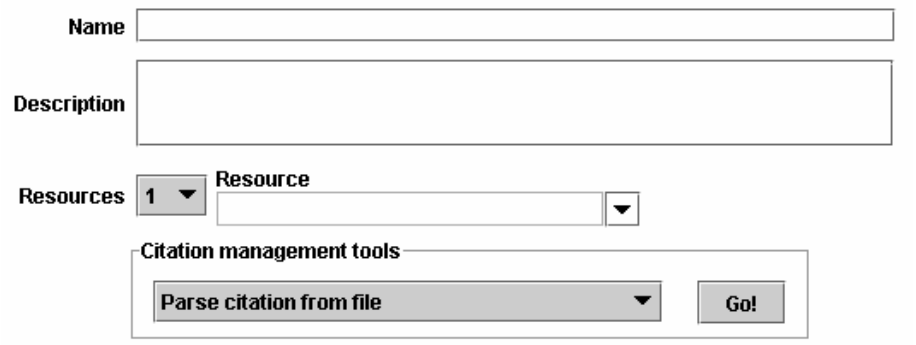
8. Select one of the references and click ‘OK’. The system will load all the reference information into the form.
9. Click the ‘commitInsert’ button to enter this reference into the knowledge base. The view will change into the ‘Display’ state (with grey letters).

8.2. Organize papers using a ResourceCollection

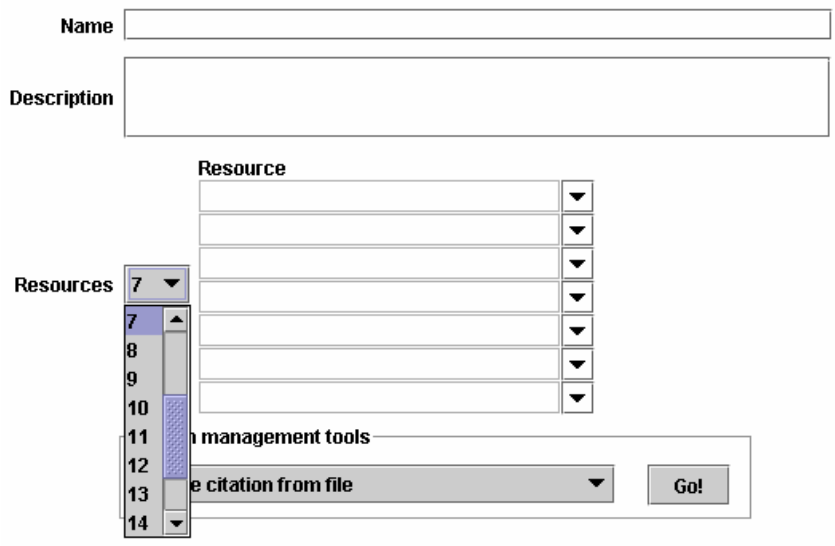
A ResourceCollection view can permit users to organize their resources according to any criteria. You could put all references pertaining to a specific topic in a collection and then create another collection for another topic. Unlike directories on your hard-drive, a single paper can happily co-exist in two collections with no difficulty.

1. Load the required knowledge base
2. Login to the knowledge base (either as root, or as a user you have already created in the system).

3. Select ‘ResourceCollection’ from the list of Views to view the Documentation for these views.
4. Click the ‘startInsert’ button. You will see a form with the following design:



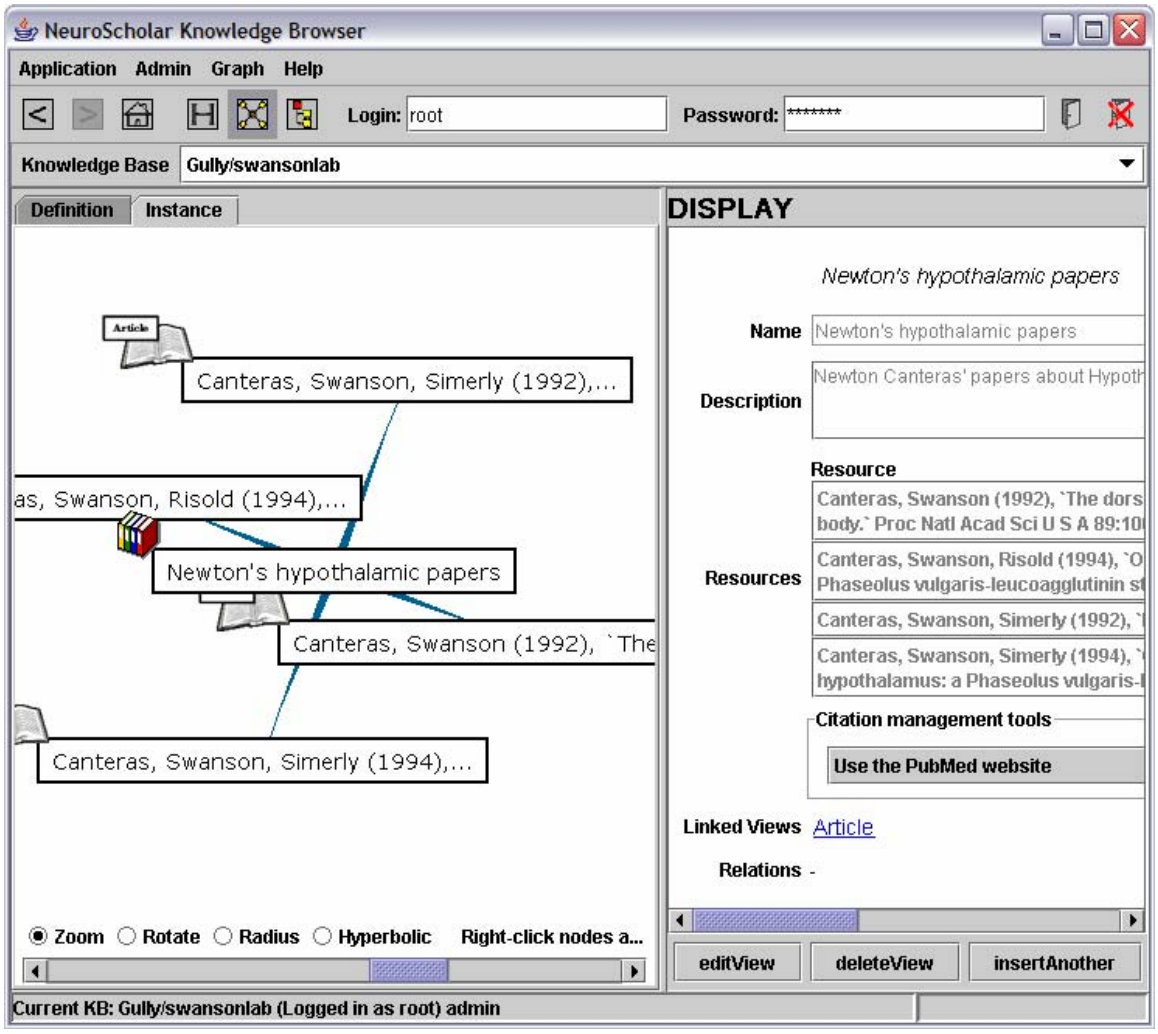
5. Pick a name for your collection; ‘connectivity studies’, ‘Norepinephrine injection studies’ or ‘Papers by my boss’ are all perfectly valid collections to use.
6. Enter a short description of the criteria that specify why you are putting these papers into this collection.
7. Using the pull-down next to ‘Resources’ select the number of resources you want to include. This could be arbitrarily high, see below.



8. In each box of this list, type the wildcard character (‘%’), followed by the name of one of the authors (or a distinctive word from the title). Note that

the paper you want to add to the ResourceCollection must already exist in the database.

9. The lookup box will return a list of entries, select the desired citation and move to the next one.

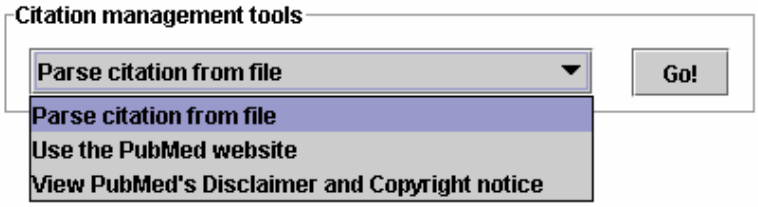


10. To confirm the addition of this ResourceCollection to your KB, click on ‘commitInsert’. The system will insert the view and revert to Display mode. An example of a small ResourceCollection is shown (with links in the view instance graph).

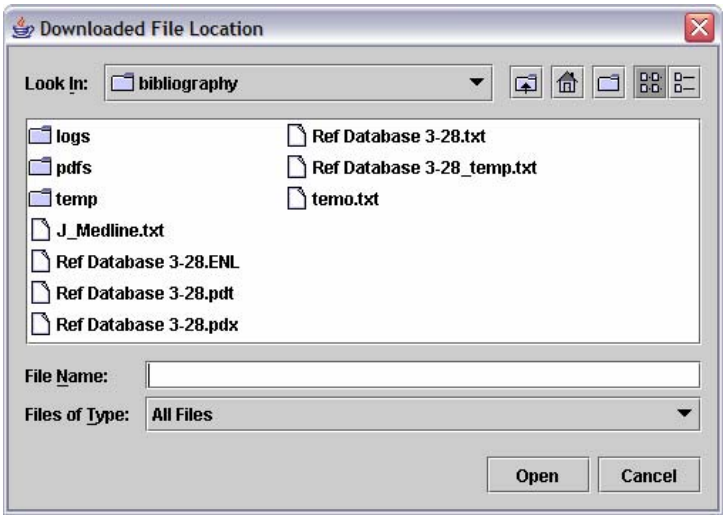
8.3. Upload a library of citations from an EndNote, Reference Manager or Procite library

Users may have large existing libraries of references that they do not want to have to go through individually uploading references into NeuroScholar. We permit these references to be uploaded as a batch file. You will need a copy of EndNote to perform this task.

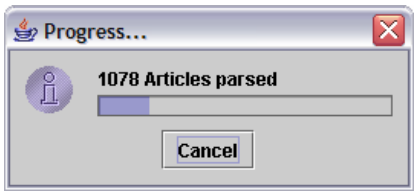
1. Repeat the steps required to insert a new ResourceCollection: load the required knowledge base
2. Login to the knowledge base (either as root, or as a user you have already created in the system).
3. Select 'ResourceCollection' from the list of Views to view the Documentation for these views.
4. Click the 'startInsert' button.
5. At this point, if your database is based on Reference Manager or ProCite, convert it to EndNote using the mechanisms within the reference management software.
6. Open the file in EndNote and export it to a file using EndNote's 'File > Export…' function.
7. Select the option 'Parse citation from file' (shown below) from the Citation Management control.



8. An 'file open' dialog then opens. Select the file of interest and click 'Open'.

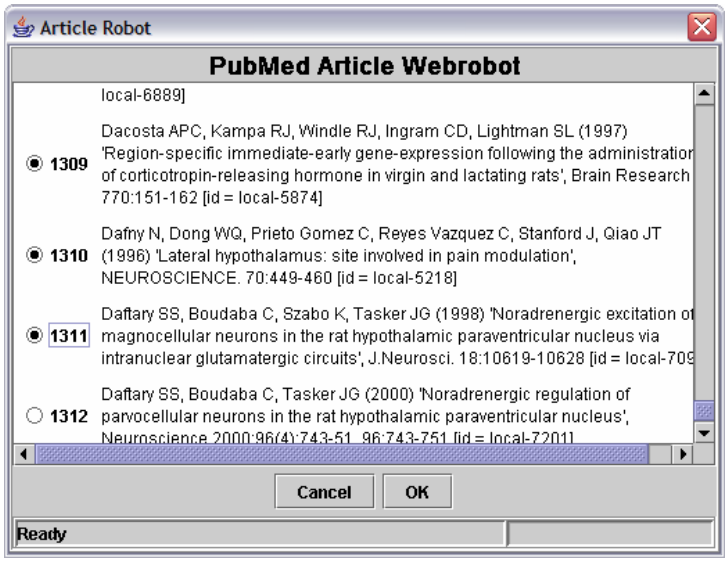


9. A progress dialog will open as NeuroScholar parses the file.



With a particularly large database, this may overload the system. If this is the case, click Cancel. The system will continue the next step and the remaining text in the reference file will be saved to disk by adding `'*_temp.txt'` to the file name.

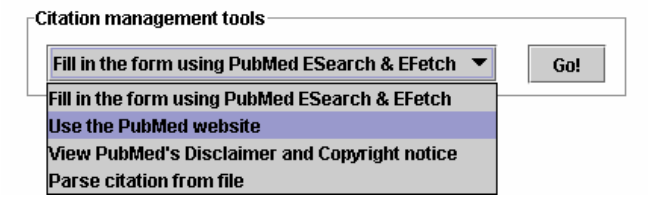
10. In *either* case, the system will display the following panel. You may select multiple entries for all the citations you wish to load and click 'Ok'. The system will then insert all selected references into the knowledge base.


Article Robot
PubMed Article Webrobot
local-6889]
1309
Dacosta APC, Kampa RJ, Windle RJ, Ingram CD, Lightman SL (1997) 'Region-specific immediate-early gene-expression following the administration of corticotropin-releasing hormone in virgin and lactating rats', Brain Research 770:151-162 [id = local-5874]
1310
Dafny N, Dong WQ, Prieto Gomez C, Reyes Vazquez C, Stanford J, Qiao JT (1996) 'Lateral hypothalamus: site involved in pain modulation', NEUROSCIENCE. 70:449-460 [id = local-5218]
1311
Daftary SS, Boudaba C, Szabo K, Tasker JG (1998) 'Noradrenergic excitation of magnocellular neurons in the rat hypothalamic paraventricular nucleus via intranuclear glutamatergic circuits', J.Neurosci. 18:10619-10628 [id = local-709
1312
Daftary SS, Boudaba C, Tasker JG (2000) 'Noradrenergic regulation of parvocellular neurons in the rat hypothalamic paraventricular nucleus', Neuroscience 2000;96(4):743-51. 96:743-751 [id = local-7201]
Cancel
OK
Ready

8.4. Add a PDF file to the reference

1. Start by inserting an article reference from PubMed as described in the How-To described above.
2. Before inserting the article into the KB, examine the entry for the Full-Text URL. If an entry exists there, click the ‘Open Link…’ button.

If no entry exists there, select ‘Use the PubMed website’


Citation management tools
Fill in the form using PubMed ESearch & EFetch
Go!
Fill in the form using PubMed ESearch & EFetch
Use the PubMed website
View PubMed's Disclaimer and Copyright notice
Parse citation from file

Both options will open a web-browser at a place where it should be relatively straightforward to navigate to the full-text article. Save this article to your hard disk in a known location.

3. Click the ‘Import…’ button of the Full Text File control. This will open a ‘File Open’ Dialogue box. Locate the PDF file that you have just downloaded. This will rename the file and upload it to the local directory in your system. The file naming scheme is dependent on the type of reference, but for Articles, it is based on the name of the first author, the year, the volume number and the start page.

At this point the PDF is now loaded into the system.

8.5. Locate an Article in the system

The system offers several ways to find specific views in the system. The easiest way is to run a query.

1. Load the Knowledge Base

2. Click ‘Article’
3. Click the ‘startQuery’ button
4. Enter some selective criteria on the form
5. Click the ‘executeQuery’ button
6. Locate the desired article in the list and click the ‘selectForDisplay’ button to display the reference.

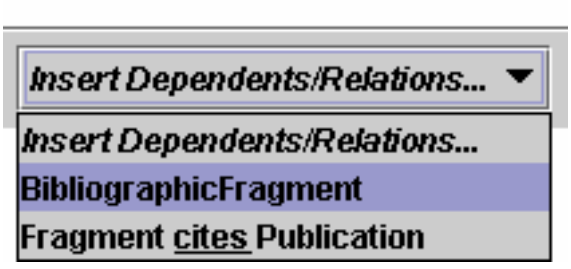
6a. If you can’t find the reference in the list, scroll down to the end and look for more pages.

6b. If you still can’t find the reference, hit the back button in the toolbar and change your query criteria.

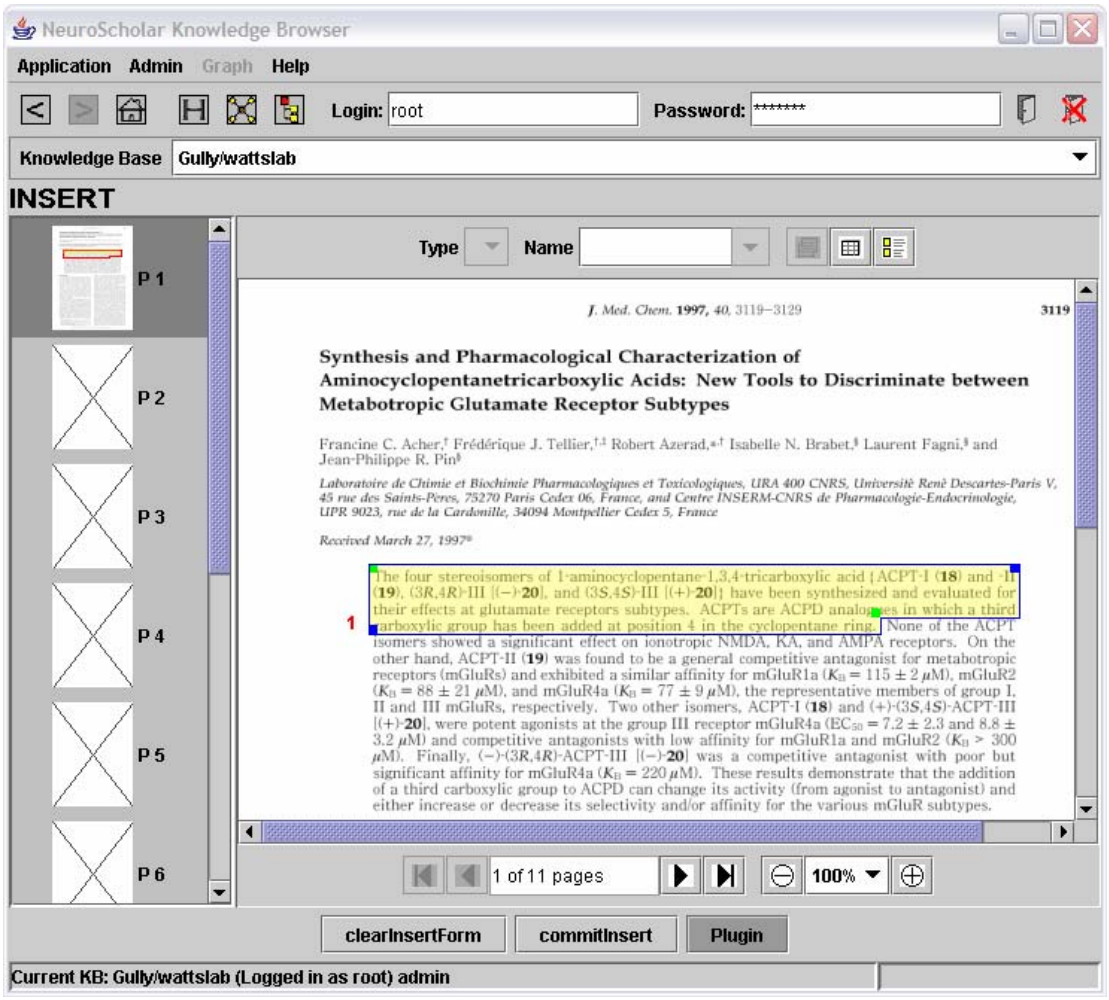
8.6. Fragment a PDF file

The primary functionality of this ‘core’ release of the software is to permit users to build a repository of searchable fragments derived from published papers. Thus inserting fragments derived from PDF files is essential for this process. Note that this section of the system uses the Multivalent java library to perform the essential parsing and rendering of the PDF file.

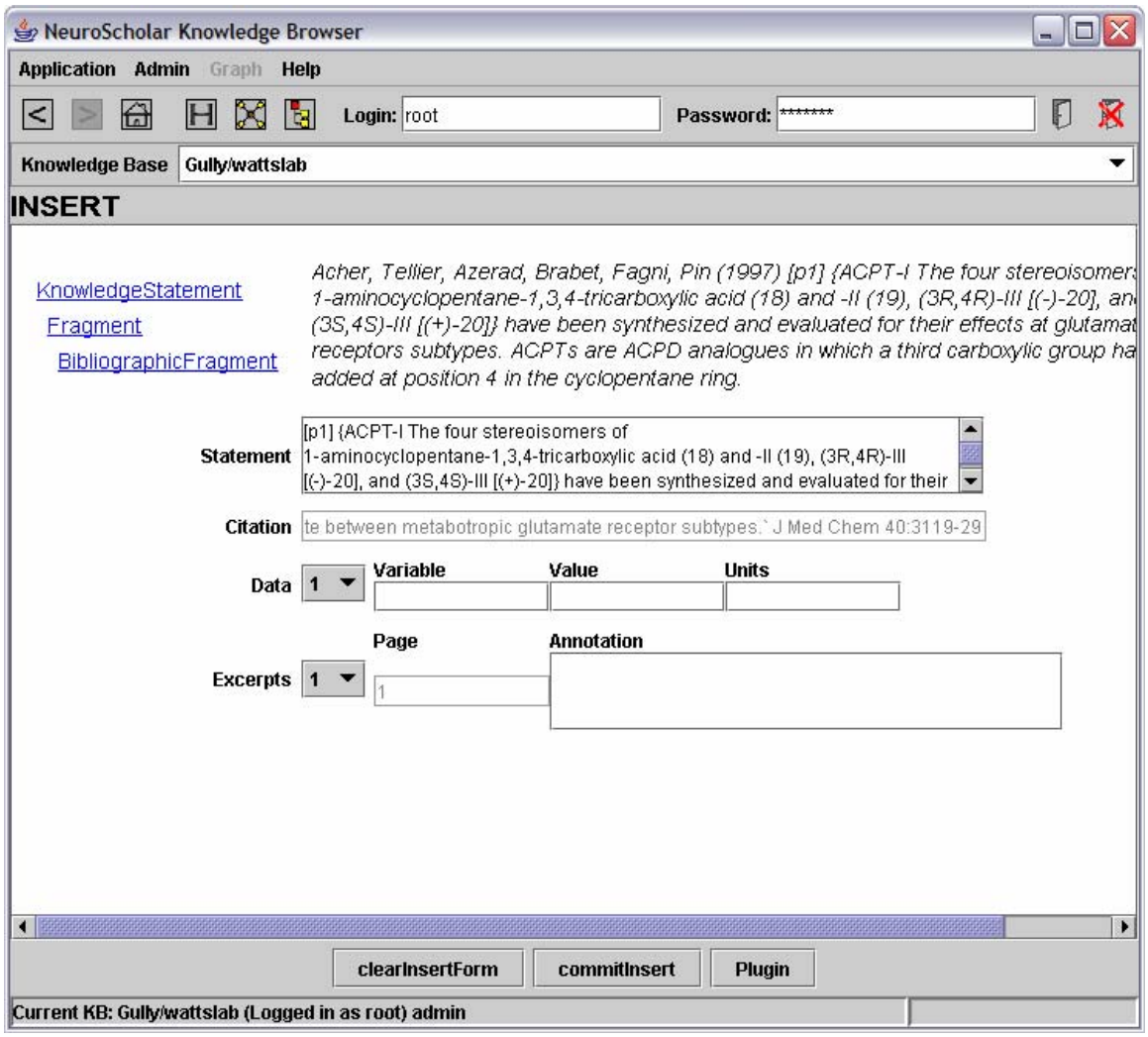
1. Start by either inserting an article with a PDF file or query the KB to show a specific Article.
2. Select ‘BibliographicFragment’ from the ‘Insert…’ pull-down control at the bottom of the form



3. The system loads the file and then presents the user with the following screen


NeuroScholar Knowledge Browser
Application Admin Graph Help
Login: root
Password: *******
Knowledge Base Gully/wattslab
INSERT
Type
Name
P 1
P 2
P 3
P 4
P 5
P 6
J. Med. Chem. 1997, 40, 3119−3129
3119
Synthesis and Pharmacological Characterization of Aminocyclopentanetricarboxylic Acids: New Tools to Discriminate between Metabotropic Glutamate Receptor Subtypes
Francine C. Acher,† Frédérique J. Tellier,†,‡ Robert Azerad,*,† Isabelle N. Brabet,§ Laurent Fagni,§ and Jean-Philippe R. Pin§
Laboratoire de Chimie et Biochimie Pharmacologiques et Toxicologiques, URA 400 CNRS, Université René Descartes-Paris V, 45 rue des Saints-Pères, 75270 Paris Cedex 06, France, and Centre INSERM-CNRS de Pharmacologie-Endocrinologie, UPR 9023, rue de la Cardonille, 34094 Montpellier Cedex 5, France
Received March 27, 1997®
1
The four stereoisomers of 1-aminocyclopentane-1,3,4-tricarboxylic acid [ACPT-I (18) and -II (19), (3R,4R)-III [(−)-20], and (3S,4S)-III [(+)-20]] have been synthesized and evaluated for their effects at glutamate receptors subtypes. ACPTs are ACPD analogues in which a third carboxylic group has been added at position 4 in the cyclopentane ring. None of the ACPT isomers showed a significant effect on ionotropic NMDA, KA, and AMPA receptors. On the other hand, ACPT-II (19) was found to be a general competitive antagonist for metabotropic receptors (mGluRs) and exhibited a similar affinity for mGluR1a ($K_B = 115 \pm 2\ \mu M$), mGluR2 ($K_B = 88 \pm 21\ \mu M$), and mGluR4a ($K_B = 77 \pm 9\ \mu M$), the representative members of group I, II and III mGluRs, respectively. Two other isomers, ACPT-I (18) and (+)-(3S,4S)-ACPT-III [(+)-20], were potent agonists at the group III receptor mGluR4a ($EC_{50} = 7.2 \pm 2.3$ and $8.8 \pm 3.2\ \mu M$) and competitive antagonists with low affinity for mGluR1a and mGluR2 ($K_B > 300\ \mu M$). Finally, (−)-(3R,4R)-ACPT-III [(−)-20] was a competitive antagonist with poor but significant affinity for mGluR4a ($K_B = 220\ \mu M$). These results demonstrate that the addition of a third carboxylic group to ACPD can change its activity (from agonist to antagonist) and either increase or decrease its selectivity and/or affinity for the various mGluR subtypes.
1 of 11 pages
100%
clearInsertForm
commitInsert
Plugin
Current KB: Gully/wattslab (Logged in as root) admin

4. You can select text (as shown above) to delineate a passage of interest. The blue points of the yellow polygon control the size of the most outward points of the polygon. The green points move inside the polygon allowing you to indent the shape to perfectly fit a small paragraph or sentence of text.
5. You may add multiple excerpts.
6. Drag and drop the excerpts to move them around the document.
7. Right click an excerpt and select ‘Delete Excerpt’ to delete the excerpt.
8. Click the ‘Plugin’ toggle button, this shows the data associated with this fragment. When no annotations are entered, the statement that is entered into the system is the enclosed text delineated within the paper itself.



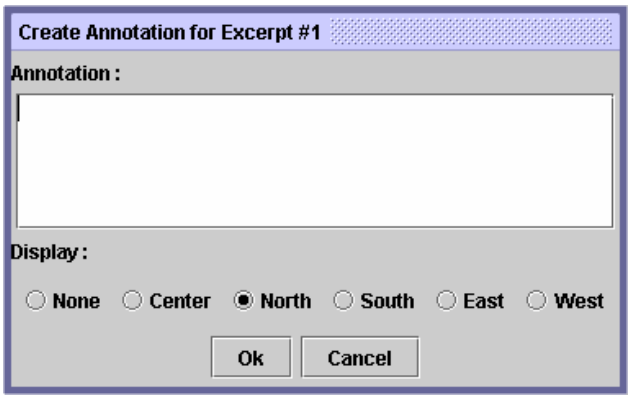
6. Click the ‘commitInsert’ button to upload the fragment into the knowledge base.

8.7. Insert a fragment with annotations (either text, voice)

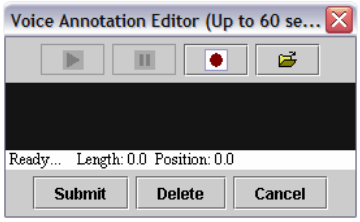
1. Follow the steps in the How-To called ‘Fragment a PDF file’.
2. After selecting a polygon, right click the yellow excerpt. Select Annotate and then either ‘Text’ or ‘Voice’.

3a. The text annotation dialog has a space in which you can enter free text and you can select where you’d like the annotation to appear on the page. These options include ‘None’ to keep the annotation hidden, ‘Center’ to place

the annotation inside the excerpt delineation or compass points to place the annotation outside the delineation at the appropriate place.



3b. The voice annotation dialog is a recorder box. Hit the record button and speak. A line will appear in the black region to show a sound trace. When you are finished, click stop and 'Submit'.



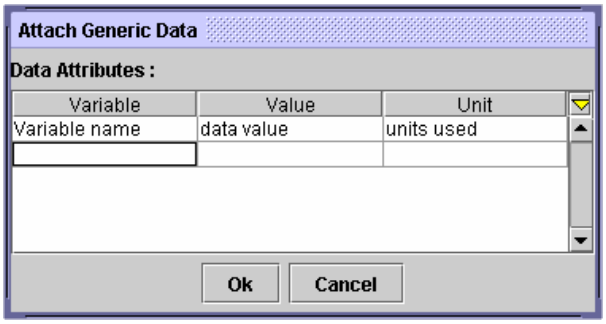
4. If you wanted to make an annotation on an excerpt act as the 'Knowledge Statement' of this fragment, right click the excerpt in question, select 'Annotate' and then 'Set as Knowledge Statement'. The annotation will then be underlined if it is displayed on the page.

8.8. Insert a fragment with data annotations

1. Follow the steps in the How-To called 'Fragment a PDF file'.
2. After selecting a polygon, click the 'Attach Data to Fragment' button on the tool bar of the fragmenter plugin. This button looks like this:

3. The dialog box used to enter variables is shown below and is essentially a spreadsheet where you may list variable names (such as 'concentration'

or some other data variable), the value and the units used. When entering these values you should try to be consistent between different fragments in terms of how you name variables.



4. Hit the ‘commitInsert’ button to upload this fragment into the KB.

8.9. Display a fragment

As with the How-To concerned with ‘locating an Article’, the easiest way to display a fragment is to run a query.

1. Load the Knowledge Base
2. Click ‘BibliographicFragment’
3. Click the ‘startQuery’ button
4. Enter some selective criteria on the form
5. Click the ‘executeQuery’ button
6. Locate the desired fragment in the list and click the ‘selectForDisplay’ button to display the reference.

6a. If you can’t find the correct fragment in the list, scroll down to the end and look for more pages.

6b. If you still can’t find the fragment, hit the back button in the toolbar and change your query criteria.

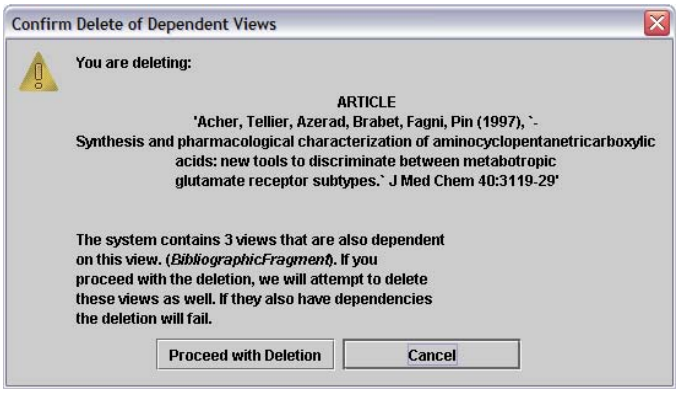
8.10. Delete a fragment

1. Use the How-To above to display the fragment you want to remove.

2. Click the ‘deleteView’ button.

8.11. Delete an article with all contained fragments

1. Use the How-To above to display the fragment you want to remove.
2. Click the ‘deleteView’ button.
3. If there are fragments associated with this paper, then the following dialog will appear:

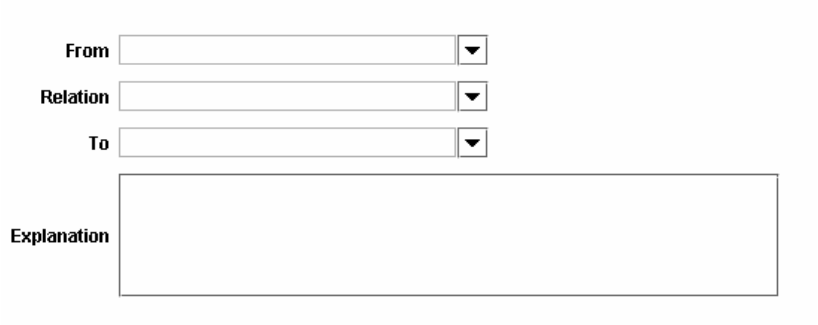

Confirm Delete of Dependent Views
You are deleting:
ARTICLE
'Acher, Tellier, Azerad, Brabet, Fagni, Pin (1997), `-
Synthesis and pharmacological characterization of aminocyclopentanetricarboxylic
acids: new tools to discriminate between metabotropic
glutamate receptor subtypes.` J Med Chem 40:3119-29'
The system contains 3 views that are also dependent
on this view. (BibliographicFragment). If you
proceed with the deletion, we will attempt to delete
these views as well. If they also have dependencies
the deletion will fail.
Proceed with Deletion
Cancel

To delete everything, click the ‘Proceed with Deletion’ button.

8.12. Linking up fragments with ‘supports’, or ‘contradicts’ relations

An important function of this system is to permit users to construct an argument from fragments derived from the primary literature. You can do this in several ways.

1. Load the KB
2. Select either ‘supports’, ‘isabout’ or ‘contradicts’ from the KB’s homepage.
3. From the Documentation page, click the ‘startInsert’ button.
4. The following empty form will appear.



5. Using the lookup boxes, locate the fragments you wish to join as well as a relation value if you wish to enter that as well.
6. Enter an explanation for the relation.
7. Click ‘commitInsert’ to upload this relation into the knowledge base.

8.13. Inserting unattached Knowledge Statements

1. Load the KB
2. Select ‘KnowledgeStatement from the KB’s homepage.
3. From the Documentation page, click the ‘startInsert’ button.
4. Enter the statement with any required variable names and values.
5. Click the ‘commitInsert’ button to upload this statement to the knowledge base.

9. Additional Support

The NeuroScholar project page on Sourceforgeprovides access to the developers in the system. If you encounter problems, please let us know by submitting a bug report or email us. (http://www.sourceforge.net/projects/neuroscholar)